PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 7 : G06F 1/00, 17/30

A2

(11) International Publication Number: WO 00/60435

(43) International Publication Date: 12 October 2000 (12.10.00)

(21) International Application Number: PCT/US00/09265

(22) International Filing Date: 7 April 2000 (07.04.00)

(30) Priority Data:
60/128,219 7 April 1999 (07.04.99) US

(71) Applicant: RENSSELAER POLYTECHNIC INSTITUTE [US/US]; 110 8th Street, Troy, NY 12180-3590 (US).

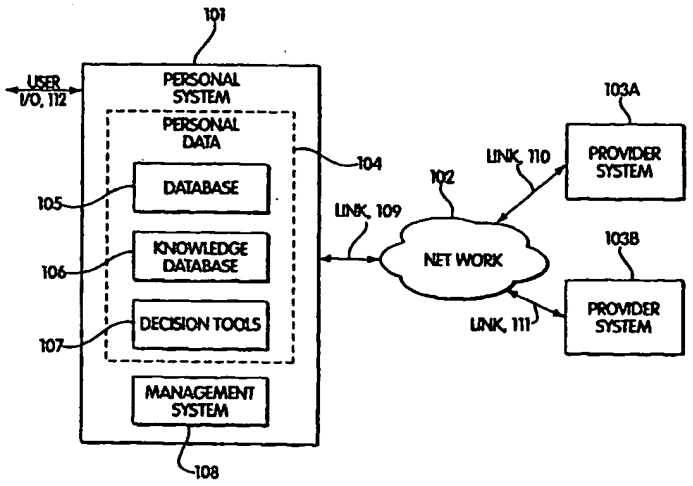
(72) Inventors: HSU, Cheng; 7 Dalamar Court, Latham, NY 12110 (US). HUGHES, Gregory, N.; 25 Old Farm Road, Chester, NJ 07930 (US). SZYMANSKI, Boleslaw; P.O. Box 301, Newtonwille, NY 12128 (US).

(74) Agent: RUSSAVAGE, Edward, J.; Wolf, Greenfield & Sacks, P.C., 600 Atlantic Avenue, Boston, MA 02210 (US).

(81) Designated States: CA, JP, European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published
Without international search report and to be republished upon receipt of that report.

(54) Title: SYSTEM AND METHOD FOR ACCESSING PERSONAL INFORMATION



(57) Abstract

A business model is provided wherein a consumer is in control of their own information. Consumer data may be controlled, for example, by a single personal system or by a plurality of systems acting in the interest of the person. A method is provided for differentiating between ownership and control of data, allowing more flexibility in distributing and controlling data. Further, a distributed system architecture is provided to distribute consumer data among a portable device and a plurality of general-purpose computer systems. The system may employ agents that perform transactions on behalf and in the best interests of the user while keeping personal data within the ownership and control of the person. A manager is provided to manage agents created by the system. In one aspect, a metadatabase is used to manage these agents. The portable device may also include a natural language user interface (NLUI or NLI) to enhance the ease of use. The portable device may use, for instance, a natural language query processor that minimized storage and processing required on the portable device.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece			TR	Turkey
BG	Bulgaria	HU	Hungary	ML	Mali	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MN	Mongolia	UA	Ukraine
BR	Brazil	IL	Israel	MR	Mauritania	UG	Uganda
BY	Belarus	IS	Iceland	MW	Malawi	US	United States of America
CA	Canada	IT	Italy	MX	Mexico	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NE	Niger	VN	Viet Nam
CG	Congo	KE	Kenya	NL	Netherlands	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NO	Norway	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	NZ	New Zealand		
CM	Cameroon			PL	Poland		
CN	China	KR	Republic of Korea	PT	Portugal		
CU	Cuba	KZ	Kazakstan	RO	Romania		
CZ	Czech Republic	LC	Saint Lucia	RU	Russian Federation		
DE	Germany	LI	Liechtenstein	SD	Sudan		
DK	Denmark	LK	Sri Lanka	SE	Sweden		
EE	Estonia	LR	Liberia	SG	Singapore		

SYSTEM AND METHOD FOR ACCESSING PERSONAL INFORMATION

RELATED APPLICATIONS

This application claims the benefit under Title 35 U.S.C. §119(e) of co-pending U.S. Provisional Application Serial No. 60/128,219, filed April 7, 1999, entitled "SYSTEM AND METHOD FOR ACCESSING PERSONAL INFORMATION" by Cheng Hsu, Gregory Hughes, and Boleslaw Szymanski, the contents of which are incorporated herein by reference.

BACKGROUND OF THE INVENTION

Field of the Invention

The present invention relates generally, to computer transaction systems, and more specifically, to systems for accessing and storing personal information.

Background of the Invention

People referred to as consumers currently conduct transactions over networks such as the Internet. These transactions are commonly referred to as electronic commerce transactions, also known as "e-business" or "e-commerce transactions." Individual entities such as companies commonly use the Internet to conduct regular business transactions including advertising their products to customers and processing transactions. Transaction information is generally stored at the company location or on the company system. If the person chooses to conduct business with another company, that person is generally required to register his or her information with that company.

It should be understood that e-commerce transactions may be performed between any entities, whether they are individuals or companies. The term "company" will be used synonymously with "provider" or "individual" providing goods or services; the company, provider or individual may be any entity providing goods or services. The terms "customer," "consumer," and "person" will also be used interchangeably as an entity that receives goods or services.

These companies also maintain information regarding consumer's behavior, and use that information to target marketing ads to specific consumers. Companies also use this information to determine what services and products should be offered

electronically, and to develop services and products. Further, these companies may also sell information regarding the consumer and their behavior to other companies.

For example, on the Internet, consumers such as people perform transactions, wherein personal information such as phone numbers, credit cards, home addresses are transferred to a provider system. The details of the transaction itself may be monitored and stored by the provider system. This provider system may use or sell this information for the benefit of itself or another company. For example, a provider system such as an e-commerce Internet site may store transaction information, such as an item type purchased by a consumer, and display items that are related to the item type to entice the consumer to perform a subsequent purchase. Or, the provider system may sell this transaction information to other companies so that these other companies may more effectively market their goods and services.

Personal information may also be provided to companies by externally-generated programs referred to as "cookies." A cookie is typically stored on a person's computer such as a personal computer (PC), and the cookie transmits information regarding the person's behavior or other information to the company. The cookie may also store information to facilitate a transaction, such as a username and password associated with the consumer.

If a user does not want a cookie to transmit personal information, the person can, at their PC, disable the acceptance of cookies. This disabling is usually performed by adjusting a setting of an Internet browser program, as cookies are generally downloaded to a person's PC while browsing the Internet or performing e-commerce transactions. When cookies are rejected, however, useful information such as usernames and passwords are not available to the user. Because this information is not available, a transaction such as an e-commerce transaction or a browsing of a Web page requires this information to be entered each time the transaction is performed, and thus the user is inconvenienced.

Some e-commerce Internet sites, such as eBay (http://www.ebay.com) and Amazon.com (http://www.amazon.com) maintain information regarding other consumer's transactions. For example, Amazon.com ranks books according to sales volume, and includes the capability of displaying feedback from other consumers. Specifically, Amazon.com provides a facility for consumers to submit reviews of books for the benefit of other consumers. Also, Amazon.com maintains information regarding

cross-sales -- Amazon.com maintains and distributes information regarding other related books that other consumers have bought from the site. However, this information is only available to the company that operates the site, and if the company permits it, consumers of the site. Further, as described above, this information is used by companies such as Amazon.com for their commercial benefit, by determining, for example, banner advertisements displayed to a user or by targeting their own products and services specifically to that user.

SUMMARY OF THE INVENTION

According to one aspect of the invention, a new business model is provided wherein a consumer such as a person is in control of their own personal information. As opposed to a business-centered e-commerce approach, wherein data is stored at the company site and is used to further advance the company's financial objectives, the paradigm is changed to a person-centered approach wherein the data of the person is owned and controlled by the person. Personal data may be controlled, for example, by a single personal system, or a plurality of systems acting together in the interest of the person.

According to another aspect of the invention, a device is provided that receives and stores personal information of the user that is configured to communicate with a system to perform transactions. The device may be, for example, an electronic chip or card which controls and stores personal databases, knowledge, and decision tools pertaining to a particular aspect of life (such as finance, medical or education) or to some integration of these aspects. The device may be portable, such that a user may carry their personal information with them to assist them in their daily lives. For example, the personal information could include medical records, and the user may share his or her medical records with one or more medical establishments that the user visits. In addition, the personal device may store personal phone numbers, networks, and other information of interest to the user. The device may store information to assist in other life activities including food preparation, health care, acquiring and maintaining housing or clothing, paying bills, acquiring information, seeking advice or enjoying recreation. In contrast with conventional commerce systems, personal data is stored with the user which allows the user to more easily perform transactions.

In one aspect, the device includes an interface that transmits the personal information to a plurality of general-purpose computer systems, where the information is processed. In another aspect, the system includes an agent that is configured to contact one or more provider systems to meet the personal needs of the users. The user does not need to know the location and identity of the provider systems; the agent contacts the provider systems transparently to the user. In another aspect, a standard transaction protocol is provided that allows companies to develop applications that use and interpret the personal information. The applications may include shared databases, knowledge and intelligence that interpret data stored on the personal device to perform transactions.

In another aspect, a user may use the personal device to assist him or her in choosing products and services. The personal device may store, for example, personal information that does not need to be replicated at each of the business systems. In another aspect, the system includes a personal information management system which allows the user to review or modify their personal information on the device or to perform transactions.

These and other advantages are provided by:

A method for performing a transaction involving data provided by a first entity to a second entity over a distributed communications network, the method comprising steps of maintaining the data, wherein the data is associated with the first entity, and controlling distribution of the data to the second entity by the first entity. In one embodiment, the first entity is a person. In another embodiment, the first entity is a company. In another aspect of the invention, the second entity is a computer system controlled by at least one of the group comprising a person; an association of people; a business; and a group of businesses. According to another aspect, the first entity controls access information sufficient to perform a transaction. In another embodiment, the method includes a step of controlling ownership, by the first entity, to the associated data. In another embodiment, the data associated with the first entity includes at least one preference of the first entity. In another embodiment, the at least one preference is determined by a transaction conducted by the first entity.

In another embodiment, the method further comprises a step of collecting, by the first entity, a history of one or more transactions performed by the first entity, and

determining at least one preference of the first entity based on the history of one or more transactions. In another embodiment, the method further comprises creating, as a result of a transaction, data owned by at least one of the group including the first entity, the second entity, and both the first and second entity. In another embodiment, the method further comprises maintaining ownership information indicating ownership of the data. In another embodiment of the invention, the data is owned by both the first and second entity, and the ownership information indicates only joint ownership information. In another embodiment of the invention, the data includes at least one preference derived from data associated with the first entity. In yet another embodiment of the invention, the at least one preference is derived from data associated with other entities. In a further embodiment of the invention, the other entities are other users in a user community, and wherein the data associated with other entities includes behavior and preference data associated with the other users.

According to another aspect of the invention, a method is provided for performing transactions over a distributed communications network involving at least one person. The method comprises maintaining, by the at least one person, data owned by the at least one person; and controlling, by the at least one person, distribution of data at least one entity over the distributed network. In another embodiment, the method further comprises a step of providing a subset of the data sufficient to perform a transaction. In another embodiment, the method further comprises a step of controlling ownership of the data by the at least one person. In another embodiment, the method further comprises a step of creating, as a result of executing a transaction, data owned by at least one of the group of the person; the at least one entity; and both the person and the at least one entity.

According to another aspect of the invention, a method is provided for maintaining ownership and control of data. The method comprises steps of operating a computer system having a plurality of processes, and wherein at least one of the processes executes as a user process; indicating, for data accessed by the at least one process, ownership of the accessed data; and indicating, for the data accessed by the at least one process, control of the accessed data wherein indication of ownership and indication of control are independent. According to one embodiment of the invention, an indication of ownership includes at least one of a group including: a first person, wherein the data is personal data of the first person; a first group, wherein the first person

is a member of the first group and the first person accesses the data through the computer system; automatic transfer of ownership of data, wherein a receiver of data including at least one of the first person or first group attains automatic ownership of the data; and wherein an indication of control includes at least one of a group including: the first person; a second person, wherein the second person accesses the data using a computer system; the first group; a second group wherein the second person is a member of the second group and accesses the data through the computer system; and a system. According to another embodiment of the invention, the indication of ownership further comprises indicating that the data has no owner. According to another embodiment of the invention, at least one of the second person or the second group, upon receiving the data, attains ownership of the data. According to another embodiment of the invention, at least one of the second person or second group attains ownership of the data based upon a predetermined relationship between an owner of the data and the at least one of the second person or the second group. According to another embodiment of the invention, the predetermined relationship is an employer-employee relationship, wherein the owner is an employee, and the at least one of the second person or the second group is an employer. According to yet another embodiment of the invention, the predetermined relationship is a legal relationship that obligates the owner of the data to relinquish ownership of the data to the at least one of the second person or the second group.

According to another aspect of the invention, a method is provided for maintaining ownership and control of data. The method comprises steps of (a) operating, by a person, a computer system, wherein the computer system is configured to operate upon the data; (b) indicating ownership of the data; and (c) indicating, independently from (b) an indication of control of the data. According to another embodiment of the invention, the steps (b) and (c) both include indicating, in the data, ownership and control of the data. According to another embodiment of the invention, the method further comprises transferring the data between first and second entities, and, as a result of a processing of the data, creating second data having an ownership by at least one of the group including the first entity; the second entity; and both the first and second entity. According to another embodiment of the invention, the method further comprises transferring the data between first and second entities, and providing control of the data by at least one of the group including the first entity; the second entity; and both the first

and second entity. According to another embodiment of the invention, the method further comprises transferring the data between first and second entities, and wherein steps (b) and (c) both include indicating, in the data, ownership and control of the data.

According to another aspect of the invention, a portable device is provided for storing personal information of a user and conducting transactions using the personal information. The device comprises a database of personal information, and an interface that transmits the personal information to an external entity to facilitate a transaction, wherein the interface controls what personal information is transmitted. According to another embodiment of the invention, the personal information is personal preferences in goods and services, and the device interprets the personal information to determine marketing information to be presented to the user. According to another embodiment of the invention, the interface further comprises a shell that permits or denies personal data to be distributed to an external entity. According to another embodiment of the invention, the interface further comprises a shell that filters unwanted data from being received by the device. According to another embodiment of the invention, a minimum amount of personal data is distributed to conduct a transaction with the external entity. According to yet another embodiment of the invention, the shell is a software program executing on the portable device. According to another aspect, the shell is a software program executing on the portable device. According to another embodiment of the invention, the device further comprises a database of information relating to the use of the personal information. According to another embodiment of the invention, the device further comprises subsystems distributed in at least one of: (a) one or more personal computers; (b) one or more network servers; and both (a) and (b). According to another embodiment of the invention, the database of information relates to the use of the personal information and includes at least one rule governing at least one of use access, and management of the personal information. According to another embodiment of the invention, the database of information relates to the use of the personal information and further comprises rules governing distribution of the personal information to external entities. According to another embodiment of the invention, the device further comprises a plurality of agents configured to support conducting transactions between a user of the device, wherein the plurality of agents control distribution of the personal information to the external entities. According to another embodiment of the invention,

the personal device cooperatively distributes the personal information among one or more trusted systems.

According to another aspect of the invention, a method is provided for accessing personal information of a user on a personal device. The method comprises steps of establishing a communication link between the device and an external system; transferring the personal information to the external system to facilitate a transaction; storing information related to the transaction on the personal device. According to another embodiment of the invention, the personal data includes medical records, and the information related to the transaction is medical treatment information. According to another embodiment of the invention, the personal data includes financial records, and the information related to the transaction is financial transaction information. According to another embodiment of the invention, the step of transferring includes transferring a minimum amount of personal data sufficient to conduct the transaction with the external system. According to another embodiment of the invention, the method further comprises a step of selecting personal data to be transferred by the step of transferring based on a minimum set of information needed to conduct the transaction. According to another embodiment of the invention, the method further comprises a step of storing, on a second external system, overflow personal information. According to yet another embodiment of the invention, the method further comprises steps of transferring the personal information to an intermediate system and generating an agent that performs the transaction on behalf of the user.

According to another aspect of the invention, a method is provided for managing personal data. The method comprises storing, on a portable device, personal data owned by a user; and controlling, by the user, distribution of the data to other entities over a communications network, wherein the personal data is distributed among the portable device and a plurality of general-purpose computers, the general-purpose computers being used to store overflow information. According to another embodiment of the invention, at least one of the plurality of general-purpose computers is a personal computer operated by the user. According to another embodiment of the invention, the method further comprises filtering unwanted data from being received by the device. According to another embodiment of the invention, the method further comprises allowing personal data to be communicated to another entity, the communicated personal data being sufficient to support a transaction. According to another embodiment of the

invention, the communicated personal data is a minimum amount of data sufficient to support the transaction. According to another embodiment of the invention, the method further comprises denying, communication of information unnecessary for performing the transaction. According to another embodiment of the invention, the step of controlling is performed cooperatively between the portable device and general-purpose computer. According to another embodiment of the invention, the method further comprises communicating the personal data with a server system, and generating, by the server system, an agent configured to control and transmit the personal data to the other entities. According to another embodiment of the invention, the method further comprises storing, on at least one of the plurality of general-purpose computers, personal data related to a community of which the user is associated. According to another embodiment of the invention, the method further comprises providing an interoperating distributed software system, the system being distributed among the portable device and plurality of general-purpose computers. According to another embodiment of the invention, the method further comprises providing, among the portable device and plurality of general-purpose computers, a common interface to the user. According to another embodiment of the invention, the method further comprises providing, among the portable device and plurality of general-purpose computers, a common interface to application software programs. According to another embodiment of the invention, the method further comprises storing, managing, and processing rules and agents to facilitate transaction conducted by the portable device and plurality of general-purpose computers.

According to another aspect if the invention, a method is provided for managing personal data. The method comprises steps of storing, on a first system, personal data owned by a user; receiving and executing, at the first system, an externally-generated program configured to transfer information to second system; and controlling, by the user, distribution of the personal data to the second system. According to another embodiment of the invention, the externally-generated program is a software cookie. According to another embodiment of the invention, the personal data owned by a user includes behavior information of the user. According to another embodiment of the invention, the step of controlling further comprises allowing access, to the externally-generated program, to a minimum amount of personal data. According to another embodiment of the invention, the step of controlling includes allowing a user to benefit

from the externally-generated program while limiting an amount of personal data accessed by the program.

According to another aspect of the invention, a method is provided for performing a transaction in a distributed network comprising generating, by a computer system coupled to a communications network, a software agent configured to exchange data to support a transaction; and exchanging the data with one or more other systems coupled to the communications network, wherein the agent provides a minimum amount of data to support the transaction. According to another embodiment of the invention, the software agent is comprised of a plurality of base agents and the step of generating comprises generating the plurality of base agents that comprise the software agent. According to another embodiment of the invention, at least one of the plurality of base agents is generated by another base agent. According to another embodiment of the invention, the method further comprises maintaining identification of each of the plurality of base agents and managing the identifications in a database. According to another embodiment of the invention, the method further comprises managing creation and deletion of agents. According to another embodiment of the invention, the method further comprises re-using existing agents to form new agents. According to yet another embodiment of the invention, the computer system includes at least three devices functioning as a single manager of data including a portable device; a system configured to connect to the portable device and store overflow information; and a server system configured to dispatch the software agent to the one or more other systems coupled to the communication network.

According to another aspect of the invention, a method is provided for interfacing a portable device to a user. The method comprises providing a natural language query input to the user; performing, based on the input, a search of one or more language-based databases; and providing, through an interface of the portable device, a result of the search. According to another embodiment of the invention, the method further comprises a step of identifying, for the one or more language-based databases, a finite number of database objects, and determining a plurality of combinations of the finite number of database objects. According to another embodiment of the invention, the method further comprises a step of mapping the natural language query to the plurality of combinations. According to another embodiment of the invention, the step of mapping comprises steps of identifying keywords in the natural language query; and relating the

keywords to the plurality of combinations. According to another embodiment of the invention, the method further comprises a step of determining a reference dictionary comprising case information; keywords; information models; and database values. According to another embodiment of the invention, the step of mapping further comprises resolving ambiguity between the keywords and the plurality of combinations. According to another embodiment of the invention, the step of resolving includes determining an optimal interpretation of the natural language query using at least one of a group comprising rules and heuristics.

Further features and advantages of the present invention as well as the structure and operation of various embodiments of the present invention are described in detail below with reference to the accompanying drawings. In the drawings, like reference numerals indicate like or functionally similar elements. Additionally, the left-most one or two digits of a reference numeral identifies the drawing in which the reference numeral first appears.

BRIEF DESCRIPTION OF THE DRAWINGS

This invention is pointed out with particularity in the appended claims. The above and further advantages of this invention may be better understood by referring to the following description when taken in conjunction with the accompanying drawings in which similar reference numbers indicate the same or similar elements.

In the drawings,

Fig. 1 is a block diagram of a personal system in accordance with one embodiment of the invention;

Fig. 2 is a block diagram showing a personal system being distributed among a plurality of systems;

Fig. 3 is a block diagram of a general-purpose computer system upon which one embodiment of the invention may be implemented;

Fig. 4 is a block diagram of a computer system using agents;

Fig. 5 is a block diagram showing agent communication in accordance with one embodiment of the invention; and

Fig. 6 is a flow diagram of a process for performing agent communication;

Fig. 7 is a block diagram showing an agent transmission process;

Fig. 8 is a block diagram showing an example metadatabase system;

Fig. 9 is a block diagram of a personal data format in accordance with one embodiment of the invention;

Fig. 10 is a flow diagram of a method of processing data in accordance with one embodiment of the invention;

Fig. 11 is a block diagram of an example graphical user interface according to one embodiment of the invention;

Fig. 12 is a block diagram of another example graphical user interface according to one embodiment of the invention;

Fig. 13 is a block diagram of a natural language query processor according to one embodiment of the invention; and

Fig. 14 is a block diagram showing an example structure of a reference dictionary.

DETAILED DESCRIPTION

The present invention will be more completely understood through the following detailed description which should be read in conjunction with the attached drawings in which similar reference numbers indicate similar structures. All references cited herein are hereby expressly incorporated by reference.

According to one aspect of the invention, a new business model is provided wherein a consumer such as a person is in control of their own personal information. In contrast to a business-centered e-commerce approach, wherein data is stored at the company site and is used to further advance the company's financial objectives, the conventional e-commerce paradigm is changed to a person-centered approach wherein the data of the person is owned and controlled by the person. Personal data may be controlled, for example, by a single personal system, or a plurality of systems acting together in the interest of the person.

According to one aspect of the invention, a method is provided for differentiating between ownership and control of data. In particular, the producer of the data has the ability to specify a level of ownership and level of control of data to allow increased functionality in handling the data. By providing separate indications of control and ownership, control and distribution of data is more flexible. For example, ownership and control information of the data may indicate a single entity, such as a person, as being the sole owner and controller of the data. For example, it may be beneficial to provide

total ownership and control of data such as name, address, Social Security Number, etc. to the person.

Alternatively, it may be beneficial to specify a joint ownership of data, wherein the group of owners collectively owns and controls the data. Here, many entities own the data, and many entities control the data. For example, a person who is a member of an organization may choose to relinquish personal data to the organization, creating community data owned and controlled by the organization. Further, the organization collectively may choose to relinquish the community data to a larger organization, of which it is or is not a part. For example, a fraternity chapter, having a collection of members, may collectively choose to relinquish fraternity chapter data to a national fraternal organization. Alternatively, the fraternity chapter may collectively choose to relinquish chapter data to an unrelated organization.

Further, it may be beneficial to provide access to data without changing ownership of the data. More specifically, it may be beneficial to loan data to another individual or company for the purpose of providing goods or services. For example, medical data owned by an individual may be loaned to a medical professional or organization for the purposes of rendering medical care. It may be useful to track the original owner of the data, as any use of the data not in an interaction with the original owner could be tracked and reported to the owner. For example, a credit report provided by a user to obtain a mortgage may be tracked by the user to prohibit use of the report inconsistent with the user's wishes. Alternatively, after the data is used for its intended purpose, such as being used in a transaction, the data may be destroyed.

Also, it may also be beneficial to provide an automatic transfer of ownership of data. For instance, an automatic transfer could be triggered by a request by an entity that has a predetermined relationship with the entity that owns the data. For example, the owning data may be an employee, and the employer may have a superior ownership right to all data produced by the employee. Thus, the employer has a legal right to the data, and the employee is obligated to relinquish the data.

According to another aspect of the invention, a device is provided for storing, managing and controlling personal data. For example, a personal system stores, manages and controls person-owned data and information. In one aspect, a shell is provided that maintains privacy, security and control of person-owned data stored in the device. Also, the shell filters out unwanted information from other individuals and companies.

According to another aspect of the invention, a distributed system architecture is provided including a portable device and a plurality of general-purpose computer systems. One or more of the general-purpose computer systems may be used to backup data and store overflow data from the portable device. In conventional systems such as PDAs coupled to PCs, information is merely replicated from the PDA to the PC, and vice-versa. According to one embodiment, overflow data, such as data that is not currently being used, is stored on at least one of the general-purpose computer systems, which generally offer more storage capabilities than the portable device. Thus, the personal data is actively managed based on use of the data. Also, the distributed system may operate collectively as a single unit, to manage and control personal data.

According to a further aspect of the invention, the system may employ agents that are created by the system that pass through the shell and return with the desired information or complete a transaction to serve the persons interests while keeping the personal data within the ownership and control of the person.

According to another aspect, a manager is provided that manages a plurality of agents. For example, the manager may facilitate re-use of existing agents to create new agents. In one embodiment, a metadata database is used to manage the agents.

In yet another aspect, the portable device includes a Natural Language User Interface (NLUI) to enhance the ease of use. In one embodiment, the portable device uses a natural language query processor that minimizes the amount of storage and processing required on the portable device.

Fig. 1 shows a block diagram of a personal system 101 in which personal data 104 may be managed and accessed by a user. System 101 may include, for example, a portable device on which the personal information 104 is stored. The personal information 104 may be, for example, data stored in a database 105, such as identification, security, medical records, financial information, or the like. Personal information 104 may also include a knowledge database 106, which is a collection of knowledge accrued by the user of the personal system 101. Knowledge base 106 may be, for example, a collection of facts, a collection of logic to be applied to facts, or both. For example, if a person is taking a first prescription drug (a fact), the fact that another prescription drug causes an interaction with the first is a logic relation. These relations may be stored in system 101 and may also be used by provider systems 103A, 103B to facilitate transactions. Personal system 101 may also include decision tools 107 which

allow the user to facilitate transactions. For example, decision tools 107 may assist the user in selecting one product over another. Decision tools 107 may include common or generic knowledge regarding providers of goods and services, or specific personal information particular to an individual user, such as medical history, background, allergies, product or brand preferences, etc.

The personal system 101 may be portable or include some portion thereof which is portable and is associated with a user. The user may carry system 101 or a portion thereof, which may be for example, a card, an electronic chip, or other device used to store personal information. In one aspect, the personal system may be attached to the user, such as being embedded in the body of the user. System 101 may or may not be accessible through a user interface located on the device. In one aspect, the device does not provide an interface on the device itself; the device is accessed by an external reading device such as a card reader. According to one aspect, a graphical user interface is provided, through which the user provides input and receives output (user I/O 112). Other methods of accessing personal data 104 may be used.

Traditional business transactions, such ATM (automatic teller machine) transactions involve the use of traditional credit cards or what is referred to in the art as "smart cards." Smart cards function as usage keys for users to gain access to services. Generally, a smart card includes only identification and security information used to authenticate a user. By contrast, personal system 101 maintains and controls information associated with the owner of the information.

According to one aspect of the invention shown in Fig. 2, personal system 101 is distributed among a portable device 201 and a number of general-purpose computer systems 202, 203. The portable device 201 includes storage media used to store the personal data or information 104A. Storage media may be, for example, magnetic or optical storage media, or electronically programmable media such as an EEPROM or Compact flash device. It should be understood that any media type may be used. Preferably, the storage media is compact and portable by a user and is energy efficient, to reduce power requirements for system 101. System 101 enables secure identification of the user and includes identity and password information on system 101. System 101 may include a processor, operating system, and memory and storage devices (not shown) used to store, retrieve and process personal data 104. In one aspect, system 101 provides access to goods and services provided by provider systems 103A, 103B.

Personal system 101 communicates with systems to exchange the personal information to facilitate transactions with one or more provider systems 103A, 103B. The personal information is generally transmitted over links 109-111 to provider systems 103A, 103B such as Internet sites, personal computers, or other systems capable of performing transactions. Links 109-111 may be optical, electromagnetic or electrical links as is known in the art. For example, links may be wireless communication links. In another aspect, the communication links 109-111 may be network links over copper or optical media. Other link types are possible.

Personal system may communicate information to provider systems directly, or may communicate information in conjunction with one or more general-purpose computers 202, 203. Computers 202, 203 may be located, for example, in a medical office or store, or at the location of a provider of goods and services. Alternatively, computers 202, 203 may be a personal computer of a user. According to one embodiment of the invention, computer system 202 may be a personal computer that can be used to store user data that is too large to be stored at device 101. In another embodiment, computer system 203 is maintained by a trusted provider and provides service to one or more users. Unlike conventional service providers that accepts advertising revenue to target consumers and otherwise do not act entirely within the best interests of the consumer, computer system 203 can be operated by the trusted provider to serve the needs of consumers. In one aspect, computer system 203 limits the access of other entities to personal data 104. Also, computer system 203 stores community data 212 to be shared among users, and used for the benefit of the users. This contrasts to current practices such as Amazon.com, wherein the company Amazon.com maintains personal and collective knowledge of transactions to further their business objectives.

Portable device 201 and computer systems 202, 203 may include a distributed management process 108A, 108B, 108C (collectively, item 108) that manages personal data 104. Data 104 may be, for example, identification information, medical records, medical treatments, previous transactions, financial information, or other personal information used by the business to perform a transaction.

Computer system 203 communicates through network 102 with one or more provider systems 103A, 103B (collectively, item 103) to allow the user to access goods and services. Provider systems 103A, 103B may in turn access other providers, such as Citigroup or other institution for financial services, an HMO for health services, etc.

Computer system 203 may provide the interfaces necessary to communicate with provider systems, such that reprogramming of the provider systems is unnecessary. Computer system 203 may communicate with other systems using a communication protocol such as TCP/IP. It should be understood that other protocols may be used.

Computer system 203 may include one or more agents used to interpret personal information, and communicate with one or more provider systems 103. Agents, discussed in more detail below with reference to Figures 4-7, are well-known in the art of Internet programming. Computer system 203 may include an agent library 213, which includes a number of agents configured to perform specific tasks. As different tasks need to be performed, additional agents may be added to the library 213.

Portable device 201 generally includes a personal information management system 108A which allows the user to view, change and interpret his or her own personal information 104 stored on personal system 101. Portable device 201 may also be a general purpose computer system 301 as shown in Fig. 3. Portable device 201 may also be a personal digital assistant (PDA) such as those available from Palm Computing, or a PDA that executes the WinCE operating system, available from a variety of vendors. Device 201 may also by a cellular phone capable of communicating data over the Internet (termed a "Web phone"). Other portable devices may be used. Portable device 201 may also include one or more application software systems 207 that execute on device 201 and operate on personal data 104. For example, a calendar management application that provides management of appointments and dates may execute on device 201. Other application systems 207 that use personal data 104 may be used.

According to another embodiment of the invention, portable device 201 includes a graphical user interface (GUI) through which a user provides input and receives output. That is, the user provides and receives user input/output (I/O) 112 to GUIs 204, 208 as shown on Figure 2. In one embodiment of the invention, discussed below with reference to Figures 10-11, the GUI is a natural language user interface (NLI) that accepts natural language inputs from the user.

Computer systems 202, 203 may also be general purpose computer systems 301 such as that shown in Fig. 3. The computer system 301 may include a processor 302 connected to one or more storage devices 303, such as a disk drive through a communication device such as bus 307. The computer system also includes one or more output devices 304, such as a monitor or graphic display, or printing device. The

computer system 301 typically includes a memory 305 for storing programs and data during operation of the computer system 301. In addition, the computer system may contain one or more communication devices that connect the computer system to a communication network 306.

Computer system 301 may be a general purpose computer system that is programmable using a high level computer programming language. The computer system may also be implemented using specially programmed, special purpose hardware. In the computer system 301, the processor 302 is typically a commercially available processor, such as the PENTIUM, PENTIUM II, PENTIUM III, or StrongARM microprocessor from the Intel Corporation, PowerPC microprocessor, SPARC processor available from Sun Microsystems, or 68000 series microprocessor available from Motorola. Many other processors are available. Such a processor usually executes an operating system which may be, for example, DOS, WINDOWS 95, WINDOWS NT, WINDOWS 2000, or WinCE available from the Microsoft Corporation, MAC OS SYSTEM 7 available from Apple Computer, SOLARIS available from Sun Microsystems, NetWare available from Novell Incorporated, PalmOS available from the 3COM corporation, or UNIX available from various sources. Many other operating systems may be used.

The communication network 102 may be an ETHERNET network or other type of local or wide area network (LAN or WAN), a point-to-point network provided by telephone services, or other type of communication network. Information consumers and providers, also referred to in the art as client and server systems, respectively, communicate through the network 102 to exchange information.

It should be understood that the invention is not limited to a particular computer system platform, processor, operating system, or network. Also, it should be apparent to those skilled in the art that the present invention is not limited to a specific programming language or computer system and that other appropriate programming languages and other appropriate computer systems could also be used.

Fig. 4 shows a detailed block diagram of one embodiment of the invention. In one aspect, computer system 401 may be similar in function to computer system 203, having an agent library 213, and operating on personal data 104 and community data 212. System 401 includes a management system 108 used for creating, tracking, encrypting and performing various other tasks with respect to personal data 104.

Further, system 108 may be responsible for managing the library 213 of software agents that handle personal data 104. Information regarding states of these agents may, for example, be stored in an agent database 405. In one embodiment, the agent database is a metadatabase capable of storing thousands of agents.

Generally, system 401 includes a communication protocol, and processes that perform application functions including authentication, accessing and storing personal information at system 401. A communication protocol may include, for example, network layer and data link layer protocols as is known in the art for communicating information between systems. An example of a network layer protocol is TCP/IP. An example of a data link protocol is Ethernet. It should be understood that any protocols may be used to communicate information between system 401 and 402 through communication link 406. System 401 includes storage media which may be, as discussed above with reference to Fig. 3, any media used to store personal information 104. A process of system 401 may access the media to store and retrieve personal information 104.

Agent library 213 may include a number of software agents, which, in the context of various aspects of the invention, are software programs that generally have several features in common. An agent is generally described as one that acts or has the power or authority to act or represent another. A software agent is generally a program that carries out activities in a flexible and intelligent manner that is responsive to changes in an environment in which the agent exists without requiring constant human guidance or intervention. A software agent may have the following attributes as described in the book entitled Software Agents edited by Jeffrey M. Bradshaw, American Association for Artificial Intelligence, MIT Press 1997, Cambridge, MA.:

Reactivity: The ability to selectively sense and act.

Autonomy: The agent has goals, and the agent is self-directed and proactive in achieving the goals.

Collaborative behavior: The agent can work in concert with other agents to achieve a common goal.

Temporal continuity: The agent has a persistence of identity and state over long periods of time.

Adaptivity: The agent is able to learn and improve with experience.

Mobility: The agent is capable of migrating in a self-directed way from one host platform to another.

It should be understood that agents having varying degrees of these attributes may be used, and still satisfy the spirit of various embodiments of the invention. Further, there may be other methods of characterizing and implementing agents which is still within the scope and spirit of the invention. For example, agents may be implemented in a variety of computer languages, some being general-purpose languages like the Java, Javascript, C, C++, ActiveX, and the Tcl programming languages, others have been specially adapted for agent programming. Aglets available from IBM, General Magic's Telescript, are agent libraries that provide standard functions for creating agents. Other agent programming languages are available. It should be understood that any programming language may be used, including scripting or compiled programming languages that are executed, interpreted, or both. The term agent code is used herein to describe agent programs written in any programming language.

Agents are particularly suited for certain tasks that require mobility, are time consuming, are repetitive, and/or require a large degree of connectivity between a system and provider. For example, searching and retrieving Internet information is a task that is well-suited for an agent. The agent, while being executed on a provider system, may reduce the amount of processing required at the requesting system and may reduce the amount of information transferred over the Internet. Further, multiple agents may be dispatched that perform transactions in parallel, thus increasing performance of the requesting computer system. Other applications of agents are possible, especially when a program is needed that meets one or more of the agent attributes described above.

As discussed, agents may be mobile in that they may "travel" or be transmitted from place to place, such as between computer systems. For example, system 401 may include an agent library 213 including a plurality of agents 403A-403ZZ that may be transmitted to a provider system 402. These agents may operate on data 104, 212, and/or transfer this data to the provider system 402. Agent library 213 may include, for example, a database used to track and store agents created by system 401. Agent library 213 may also include a managing agent 404 which is executed on computer system 401, and monitors a state of a query or transaction.

Provider system 402 may include an agent processor 408 that executes code of the agent, or otherwise creates an environment in which an agent may operate. This

agent processor may include, for example, an interpreter or processor that executes code statements of the agent. Using the Java programming language, agent processor 408 may include a Java Virtual Machine that interprets agent code. Also, agent processor 408 may itself be a program that interprets agents transmitted to system 402. Provider system 402 also includes agent storage 409 wherein the code associated with the agent is stored, and wherein the agent stores data created from transactions, queries, and other operations performed by the agent.

In one embodiment of an agent system, mobile intelligent agents are used. Figure 5 shows a system for communicating agents in accordance with one embodiment of the invention. Computer system 401 creates a managing agent called an initiating agent 501 that controls access to personal 104 and community 212 data, and issues agents to provider system 402 to perform one or more transactions. In one embodiment of the invention, any initiation of interaction between the personal device and one or more provider systems 402 triggers the creation a pair of mobile agents which will perform the interaction. One of the agents in this pair is the initiating agent 501 for the interaction. The other agent is a mirror agent 502 that remains at the initial server and monitors the state of the initiating agent 501. The initiating agent 501 retrieves a location of the personal data 104 and issues one or more agents to establish a connection between the memory location and the provider system 402 designated for the initiated interaction. The memory location for personal data 104 may be, for example, located on computer system 401 or distributed among one or more trusted computer systems. This provider system 402 to which agents are issued is selected by the initiating agent 501 based on the locations of personal data 104 and location of portable device 201 requesting the transaction.

The initiating agent 501 travels to the designated provider system 402 and is responsible for performing transactions on behalf of the user. For example, the user may issue a request, such as a query, through an interface 206 of the portable device 201, and system 203 issues agents to service the request. The agent 501 is transmitted to system 402, where it is stored in agent storage 409 as agent 506. According to one embodiment, the agent 501 and its information is encrypted prior to transmission over untrusted networks.

Initiating agent 506 generates query 507 and transaction 508 agents in response to the user requests and interacts with memory agents 509 located on the provider or

other system to obtain data from the designated provider system 402. By indicating as their source, the selected server and not the user site, as well as by carefully limiting the user information revealed in processing, query 507 and transaction agents 508 protect the user from revealing any information to visited web-sites and other providers 402 that is not necessary for performing an intended transaction. The initiating agent 501 process and transform results generated from query 507 and transaction 508 agents to a form suitable for the computer system 401 to process and communicate the results to the user as a part of the portable device 201 and initiating agent dialog. If initiating agent 506 does not receive response from one of the query 507 or transaction 508 agents in a pre-defined time limit, initiating agent 506 will reissue this agent one or more times to attempt to process the relevant transaction or query. The reissued transaction may include processing to avoid potential double execution of the transaction (by a lost or delayed agent and by the reissued agent). In one aspect, the reissued agent is not identical to the original agent; conventional systems typically reissue a new agent that is identical to the first agent.

The initiating agent 501 propagates any changes in the user memory of system 401 to all relevant locations by issuing update memory agents 503, and reissuing them if necessary, until all memory agents report a successful completion of the changes. Temporary agent storage 409 on the designated provider system 402 is cleared and traces of the performed transactions are removed from the system 402, if possible, to minimize the amount of personal data residing at the provider system 402. Initiating agent 506 kills all agents it generated and issues a final report to its mirror agent 502. The mirror agent 503 kills initiating agent 501 and finally, itself.

During its existence, initiating agent 506 periodically (with, for instance, a frequency defined by the system), report its status to the mirror agent 502. If the mirror agent 502 does not receive this report within a predefined time, mirror agent 502 issues a kill command to the initiating agent 506 and creates a reissued initiating agent 504 that restarts processing at the last report time. A principle difference between the reissued 504 and original 501 initiating agent is that the reissued agent 504 will generate all transactions initially as reissued to avoid duplicating transaction execution.

Because two different subsets of memory parts can be used for two different user-system interactions, processing of one of this interactions is blocked. If such situation happens, both transactions determine the lowest numbered user memory part shared

between them and determines which transaction docked a memory agent 503 at this part first. The transaction that docked the lowest-numbered shared memory part is allowed to proceed, whereas the other transaction is blocked until the consistent memory status will be achieved by the proceeding transaction.

According to one embodiment of the invention, all of the agents generated, except the mirror agent 502, are temporary, meaning that each agent ceases to exist at a predefined time to avoid double processing. This may be accomplished, for example, by associating, with each agent a "certificate of life" that specifies a preset time of existence for the agent. A managing agent such as a mirror agent 502 can extend this certificate of life by an additional time in certain situations (long transfer delays, long processing queues, etc.). For example, the mirror agent 502 may be configured to adjust the certificate of life of all other agents, and the initiation agent 501 may be capable of adjusting certificates for all other agents except itself. In this manner, agents perform their own "cleanup" processing, e.g. they do not cause a loss of working memory, while still being able to perform processing in a variety of environmental situations provided by the network, provider 402 or computer system 401.

Programs, such as cookies, can also be transferred back to computer system 401. According to one embodiment of the invention as shown in Figure 2, computer system 203 may include a storage area (not shown) for incoming agents or cookies. The storage area protects the system 101 from unauthorized access to personal 104 and community 212 data, while allowing the user to have the benefit of using cookies. In conventional systems, if a user desires a higher level of security, the conventional system may be configured to block accepting cookies. However, cookies may include useful information such as usernames and passwords to be used at the provider systems with which they are associated. To solve this problem, system 101 allows the acceptance of cookies, but does not allow a cookie unrestricted access to the user's personal 104 and community 212 data. This may be accomplished, for instance, by executing the cookie as a restricted user, wherein data 104, 212 is stored in a protected storage area not accessible to the cookie. Alternatively, data 104, 212 may be stored on another system, such as device 201 or computer system 202, and transferred to system 203 in a secure manner only when needed. The user operates device 201 which transmits a request (such as a query or other transaction) to system 203, wherein system 203 performs the transaction on behalf of the user. Because the system 203 has access to the cookie, the

system is capable of providing the benefit of the cookie to the user, while not divulging data 104, 212.

Collectively, the system provides privacy, security, fault tolerance and efficiency of transaction processing in the distributed collection of the secure servers and the distributed memory consistency for the user memory. The query 507 and transaction 508 agents provide user privacy by disassociating the user from transactions. Memory agents 503, initiating agent 501 and query 507 and transaction agents 508 provide communication security by carrying only encrypted data over untrusted networks and by decrypting them only at the point of processing. Because processing of data is performed on secure servers owned by the system owner, security is maximized. Thus, a protective "shell" is created at system 401, wherein personal data 104 is allowed to leave the shell by the will of the user. Personal data, when it leaves the shell, is accompanied by an agent configured to manage and protect the data. Also, system 101 and data 104, 212 are protected from external programs such as cookies.

In one embodiment, the initiating agent 501 supports processing mobility and efficiency by selecting the designated provider system 402 in accordance with the current network traffic and load on the systems. Also, mirror agent 502 provides fault tolerance by taking over the initiating agent 501 in case the initiating agent 501 fails or an unacceptable delay is experienced. Similarly, the initiating agent 501 reissues query 507 or transaction agent 508 that fails to return results in the predetermined time, thereby increasing fault tolerance of the entire system. Memory consistency is ensured by delaying any operations that result from attaching more than one memory agent 503 to any memory part. Also, the agents are temporary agents that kill themselves after the predefined time, thus conserving distributed memory resources and minimizing the amount of management required at systems 401, 402.

Figure 6 shows one embodiment of a process for performing agent communication in the system shown in Figure 5. At block 601, process 600 begins. At block 602, computer system 401 creates initiating agent 501 and mirror agent 502. Initiation agent 501 retrieves a location of data 104 at block 603, and, at block 604, initialization agent 501 issues an agent 506 to provider 402. At block 605, the issued agent 506 creates, at provider 402, transaction 508 and query 507 agents. The agents 507, 508, at block 606, perform a transaction on behalf of the user. At block 607, initialization agent 501 determines whether the query 507 and transaction 508 agents

respond within predetermined time limits. If so, the initialization agent communicated the changes to a memory of system 401 via memory agents 503. If agents 507, 508 have not replied within the predetermined time limits, initialization agent 501 generates a reissue agent at block 610. At block 611, the reissue agent issues an agent to system 402. As discussed above, the reissue agent is preferably not identical to the initialization agent 501, as the reissue agent completes the transaction at block 613 from a point last reported by the initialization agent 501. That report may be an interim report provided by initiating agent 501 to mirror agent 502. The reissue agent sent to system 402 also generates query and transaction agents at block 612. These agents also re-execute the transaction, at the point last reported.

Figure 7 shows a block diagram depicting an agent transmission process. As discussed above, an agent 701 can be mobile, and can be transmitted to a provider system 402 to perform one or more transactions. Agent 701 obtains personal data 104 needed to perform a transaction, and the agent along with data 104 is encrypted prior to being transmitted 704 over a network 102. As discussed, network 102 may be an untrusted network such as the Internet. Encryption may include, for example, private or public key encryption, a combination thereof, or any encryption method suitable for protecting data. Transmission 704 may entail, as is known in the art, transmitting the encrypted data in one or more messages according to a data transmission protocol, such as TCP/IP. Also, TCP/IP packets may be transmitted over any number of media types such as fiber or copper cabling, or wireless communication media, or any media suitable for transferring information. Decryption 705 is performed at the provider system 402, wherein the agent is executed 706. A provider process 215 services the transaction, wherein data is transferred to and from the provider process 215. Prior to transmission over network 102, the agent and/or its data is encrypted at block 707. The data is transmitted at block 708 back to the originating computer system 401, or is passed on to another provider system 402A, depending on the transaction type. That is, some transactions may require an agent to traverse more that one provider system. The agent and its data are decrypted at block 709, wherein the agent is executed 710. The agent is either terminated at block 711 if the transaction is completed, or is further encrypted at block 712 if the agent is transmitted over network 102 to a further provider system or back to computer system 401.

As discussed above with reference to Figure 4, system 401 may track agents using an agent library 213 having a database. According to one embodiment of the invention, the database is metadatabase, which is well-known in the art of data and knowledge management tools. Metadatabase theory is described in more detail in a number of books and publications, including the book entitled Enterprise Integration and Modeling: The Metadatabase Approach, by Cheng Hsu, Kluwer Academic Publishers, Amsterdam, Holland and Boston, Massachusetts, 1996. Also, metadatabase theory is described in the journal article by Hsu, C., et al. entitled The Metadatabase Approach to Integrating and Managing Manufacturing Information Systems, Journal of Intelligent Manufacturing, 1994, pp. 333-349. In conventional systems, metadatabase theory has traditionally been applied to manufacturing problems. A metadatabase contains information about enterprise data combined with knowledge of how the data is used. The metadatabase uses this knowledge to integrate data and support applications.

The metadatabase model as shown in Figure 8, uses a structure that shows how a metadatabase system 802 provides an enterprise information model describing data resources of globally-distributed provider systems applications 407 and their control strategy in the form of rules. These globally-distributed systems applications 407 may be executed, for example, at one or more provider systems 402 discussed above. The information model also includes knowledge regarding dynamics of information transfer such as "what and how" information is shared among local systems and under what circumstances it is used. The information model may be in the form of a metadatabase 801 having data items 804, models 805, rules 806, software resources 807 and application and user information 808.

As applied to a system for managing personal information using agents, the information model describes the global requirements of the agent system, such as core rules 806 of the interface languages and particular rules for their interoperation. For example, there may be specific rules that determine how a particular provider handles input and output from an HTML interface. The provider may also use some other type of interface, such as a command or scripting interface to exchange data. These rules can be represented in the metadatabase using its rulebase model. The model is detailed in several publications, including the journal article entitled A Rulebase Model for Data and Knowledge Integration in Multiple Systems Environments, International Journal of Artificial Intelligence Tools, Vol. 2, No. 4, 1993, pp. 485-509.

The metadatabase 801 may also include metadata about software resources 807 such as common software routines or methods shared by agents. That is, the metadatabase catalogs availability and functions of reusable software resources and tracks their use in one or more agents. In this way, changes made to software resources 807 may be promulgated to other agents that use these resources. New agents can also be created from groups of these software resources stored in the metadatabase.

In addition, the metadatabase contains metadata about applications and users 808 of the agent system. These metadata may, for example, include provider type information such as web site types of the application service providers, to assist the system in creating and managing agents that interface with the providers. The metadata may also include references to personal data, such as personal preferences and requirements, stored in one or more devices of system 101. Runtime agents can retrieve personal data 104 and use them, for example, parameters to set priority of information retrieval or transactions. The application and user portion 808 of the metadatabase 801 may specify how personal preferences and requirements of the user are incorporated into building runtime agents for specific applications.

The metadatabase links these various classes of metadata by using a metadata representation that is known and described in the book entitled Enterprise Integration and Modeling: The Metadatabase Approach cited above. Thus, various data objects and routines of persistent agents are interrelated in the context of applications and users. When a new agent is needed, the metadatabase matches the type of application with those of the persistent agents and their software resources in data objects. The metadatabase then identifies pertinent elements that the new agent can use or reuse, and determines personalized parameters to include with the agent. Thus, a structured method to manage agent resources and match them to users and provider applications is provided. In summary, the metadatabase 801 describes how various database objects and routines of agents are interrelated and how they can be reused to build new agents.

Also, as applied to a system of managing personal information using agents, the metadatabase may be used to track the state of agents that obtain information from a large number of distributed databases over the Internet. Because the metadatabase system design scales well to a large system of distributed databases, the metadatabase may be used to track the creation, deletion, status of agents. In particular, the metadatabase may store status of Internet applications that use these agents, and use this

status to facilitate creating and managing agents. Each of these applications has an associated entry in the metadatabase, which stores metadata about the application's type and its use of agents. When one or more applications are changed, the metadatabase can update a relationship between agents and applications, including deleting agents created expressly for these applications. However, the common software resources in data objects used to build the agents do not have to be deleted; only their interface configuration into some particular agents needs to be deleted. The metadatabase maintains relationships and cross references to facilitate this management of agents. The capability of flexibly managing and maintaining agents is critical to systems that need to manage and use large numbers of agents, especially agents whose relations to provider applications changes frequently.

The metadatabase is a repository of information about the structure and functions of applications with which agents operate and tasks that agents perform. The metadatabase may include, for instance, functional and informational models, databases and interface requirements of the Internet applications. The MDBMS is the user interface to the metadatabase and the database processor managing the metadatabase. The MDBMS makes it possible to develop, maintain and utilize information in the metadatabase to create runtime agents and manage persistent agents. Further details of the metadatabase system are described in the book entitled Enterprise Integration and Modeling: The Metadatabase Approach, by Cheng Hsu, Kluwer Academic Publishers, Amsterdam, Holland and Boston, Massachusetts, 1996.

Figure 9 shows a data format according to one aspect of the invention. In particular, a data format is provided that allows a user to differentiate between ownership and control of data. That is, the producer of the data has the ability to specify a level of ownership and level of control of data to allow increased functionality in handling the data. By providing separate indications of control and ownership, control distribution of data is more flexible. For example, data A (item 901) may have associated with it ownership information A (item 902) and control information A (item 903). Ownership information A may specify one or more owners of information A, and ownership information A is transmitted with the data A portion 901. By contrast, conventional systems associate a single username to a file by locating the file in a directory structure having certain ownership attributes. Once the file is transmitted, this ownership information is lost. Further, data A has associated with it control information A 903,

which indicates who is allowed control of the data. It is beneficial to provide a separate indication of control, because a provider system 402 may be allowed temporary control of the data for a transaction, but the provider is not the owner of the data (unless otherwise determined by the transaction type) and the provide system 402 should not be indicated as the owner.

It may be beneficial to specify a joint ownership of data, wherein the group of owners collectively owns and controls the data. Here, many entities own the data, and many entities control the data. It may also be beneficial to provide access to data without changing ownership of the data. More specifically, it may be beneficial to loan data to another individual or company for the purpose of providing goods or services. For example, medical data owned by an individual may be loaned to a medical professional or organization for the purposes of rendering medical care. It may be useful to track the original owner of the data, as any use of the data not in an interaction with the original owner could be tracked and reported to the owner. For example, a credit report provided by a user to obtain a mortgage may be tracked by the user to prohibit use of the report inconsistent with the user's wishes. Alternatively, after the data is used for its intended purpose, such as being used in a transaction, the data may be destroyed.

Also, it may be beneficial to provide an automatic transfer of ownership of data. For instance, an automatic transfer could be triggered by a request by an entity that has a predetermined relationship with the entity that owns the data. For example, the owning data may be an employee, and the employer may have a superior ownership right to all data produced by the employee. Thus, the employer has a legal right to the data, and the employee is obligated to relinquish the data. It should be understood that this separate indication of control and ownership may have other uses. In a similar manner, data B (item 904) may have associated with it, separate ownership B (item 905) and control B (item 906) information.

Figure 10 shows a process 1000 for of processing data in accordance with one embodiment of the invention. At block 1001, process 1000 begins. In a typical transaction involving a computer system 401 and provider system 402, system 401 transfers data owned by a user A (data A, item 901) along with its ownership 902 and control 903 information to provider system 402. At system 402, an agent interfaces with a process of the provider system, and performs a transaction using user data A at block 1003. As a result of performing the transaction, transaction data is produced at block

1004, the transaction data having its own ownership and control information generated at block 1005. Transaction data may indicate, for example, that both a user and the provider that generated the transaction data own the data. Alternatively, the ownership data may indicate that either the user or provider owns the transaction data. At block 1006, the transaction data is either stored at system 402, transferred back to system 401, or both. At block 1007, process 1000 ends.

As discussed above, system 101 may include a natural language user interface (NLUI or simply NLI), through which a user requests information and performs other transactions. For instance, the user may provide input and receive output from graphical user interfaces 1101, 1201 such as that shown in Figures 11 and 12. In one case, the interface 1101 may prompt a user with a series of questions 1102, 1104, to which the user may respond. The questions may be multiple choice question format, of which a single selection of the choices is an appropriate response. However, as shown in Figure 12, the system 101 may present a general query interface on graphical user interface 1201, through which the user may pose natural language queries or responses to questions. For example, at line 1202, system 101 prompts the user to "Please enter a search (natural language or keyword)." At line 1203, the user provides a natural language response, asking system 101 "Where is the Houston Field House at RPI located?" The natural language interface may have, associated with it, a natural language analyzer which determines the meaning of the input provided. According to one embodiment of the invention, the natural language analysis system is the system shown in Figure 13 discussed in more detail below. The natural language analysis system finds the meaning of the request and determines the correct source of the information requested. For example, system 101 may issue one or more agents to perform the request. The agents may filter and format the result 1204, and return the result 1204 to system 101.

In general, a natural language analyzer that analyzes queries (hereinafter termed a "natural language query processor") may be part of computer system 101. This query processor may perform one or more analyzing steps on a received query, which is generally a string of characters, numbers, or other items. A long-standing goal in the field of information technology is to allow humans to communicate with computer systems in the natural languages of humans. However, because of the various

ambiguous and implicit meaning found in natural language, queries are difficult for a computer system to interpret precisely.

Most of the conventional methods for understanding natural language queries involve determining a method of understanding a user's natural language articulation by implementing various methods of artificial intelligence. There are several conventional approaches to limiting the naturalness of language so that the system could reduce the complexities of interpretation and successfully process queries. They include (1) template-based approach [e.g., Weizenbaum, J. 1966, ELIZA-A Computer Program for the Study of Natural Language Communication between Man and Machine, *Communications of the ACM* Vol. 9 No. 1: pp. 36-44], (2) syntax-based approach [e.g., Waltz, D.L. 1978, An English Language Question Answering System for a Large Relational Database, *Communications of the ACM* Vol. 21 No. 7: pp. 526-539, Codd, E.F, R.S.Arnold, J-M. Cadiou, C.L. Chang, and N. Roussopoulos, RENDEZVOUS Vesion 1: An Experimental English Language Query Formulation System for Casual Users of Relational Data Bases, IBM Research Report RJ2144, 1978., Codd, E.F., How about Recently? (English Dialog with Relational Databases Using Rendezvous Version 1). In B. Shneiderman (Eds). Databases: Improving Usability and Responsiveness, 1978, pp. 3-28.], (3) semantics-grammar-based approach [e.g., Hendrix, G.G. Sacerdoti, E.D. Sagalowicz, C. and Slocum, J. 1978, Development a Natural Language Interface to Complex Data, *ACM Trans. on Database Systems* Vol. 3. No. 2: pp. 105-147], (4) intermediate-representation-language-based approach [e.g., Gross, B.J. Appelt, D.E. Martin, P.A. and Pereira, F.C.N. 1987, TEAM: an Experiment in Design of Transportable Natural-Language Interfaces, *ACM Trans*actions Vol. 32: pp. 173-243], and (5) semantics-model-based approach [e.g., Janus, J.M. 1986, The Semantics-based Natural Language Interface to Relational Databases, in L. Bolc and M. Jarke (Eds), *Cooperative Interfaces to Information Systems*, pp. 143-187. New York: Springer-Verlag Janus, J.M. 1986, The Semantics-based Natural Language Interface to Relational Databases, in L. Bolc and M. Jarke (Eds). *Cooperative Interfaces to Information Systems*. pp. 143-187. New York: Springer-Verlag, Motro, A. 1990, FLEX, A Tolerant and Cooperative User Interface to Databases, *IEEE Transactions on Knowledge and Data Engineering*. Vol. 2 No. 2: pp. 231-246, Guida, G. and Tasso C. 1982, NLI: A Robust Interface for Natural Language Person-Machine Communication, *Int. J. Man-Machine Studies* Vol.17: pp.417-433].

These conventional approaches differ in the way each controls the input and in the extent to which each exerts the control on the user. The first four approaches (1) - (4) require users to articulate only in the natural language forms that the system provides -- or at least they assume that the user's articulation is consistent with these underlying forms. When this basic requirement or assumption does not hold in practice, the system would fail to function properly (e.g., with poor performance and low accuracy), or even fail altogether. These forms typically feature some generic, linguistic prototype consisting of only one single sentence per query. Thus, their advantage is that the resultant NLI is easily portable from one database system to another. The disadvantage is the restriction on naturalness of the input from the user. The last approach (5) essentially embraces a different priority, placing naturalness ahead of portability (i.e., coupling a particular NLI design with a particular domain of application, but allowing free-format text as input). If the first four approaches are top-down in their relying on the computer's direct understanding of the user's articulation, the last one could be considered as the computer's exhausting of all possible interpretations from the bottom up.

The basic strategy of system (1) - (5) is to provide a semantic model or a dictionary as the roadmap for generating possible interpretations. These systems assume that the users always query databases known to the system, thus the NLI could be tuned according to this known information. According to one embodiment of the invention, it is recognized that, under this assumption, users are bound to refer, either directly or indirectly, to these known database objects (types or semantic models, instances or values, and operators) in their natural queries. If they do not use directly these database objects, they have to articulate their query in terms of other significant words and phrases (hereinafter referred to as "keywords") that correspond to these objects. Thus, the domain of interpretation is finite, compared to natural language processing in general. Semantic model is a form of keywords and a dictionary is a more extensive collection of keywords beyond a usual semantic model. The critical success factor of this approach is clearly the semantic model-dictionary employed, which must be powerful enough to span efficiently the space of possible usage of natural language in the domain. Because database objects can only be a grossly simplistic portion of the natural vocabulary, keywords must shoulder the burden of representing naturalness. Their number could increase exponentially as the number of users and usage patterns increase.

Figure 13 shows a natural language query processor 1301 according to one embodiment of the invention. Processor 1301 receives a natural language query 1302 and a plurality of database objects 1304A, and produces a query result 1303. The natural language query may be, for example, a paragraph, a sentence, sentence fragment, or a plurality of keywords. The query result may be any information that is relevant to the combination of database objects 1304A and query 1302. According to one embodiment, the natural language query 1302 is mapped to the plurality of database objects 1304A using a reference dictionary 1308 comprising keywords 1309, case information 1310, information models 1311, and database object values 1304B. An advantage of this mapping is that less-capable processing hardware is needed to perform the mapping than traditional natural language processing algorithms such as those cited above, because the number of keywords that needs to be recognized and searched by the system is reduced. This advantage enables, for example, use of such an NLI on portable device 201 as shown in Fig. 2. Also, because the system 201 may be allocated to a single user, and processor 1301 is capable of learning using case-based learning, processor 1301 may become more accurate for the particular user.

It should be understood that other natural language query processors may be used. In general, natural language processors are well-known, and functions they perform are described in more detail in the book entitled Natural Language Understanding, by James Allen, Benjamin/Cummings Publishing Company, Inc., Redwood City, CA, 1994, herein incorporated by reference. Other natural language query processors are discussed in the journal articles and books cited above.

According to one embodiment of the invention as shown in Fig. 13, query processor 1301 includes a reference dictionary object identifier 1305 that parses query 1302 and generates one or more objects recognized in the reference dictionary 1308. Reference dictionary object identifier 1305 also identifies words that are meaningful in the reference dictionary 1308 and eliminates useless or unmeaningful words. Processor 1301 also accepts and processes a number of database objects 1304A. As discussed, processor 1301 may have an associated reference dictionary 1308 that includes keywords 1309, case information 1310, information models 1311 and one or more database objects 1304B. Keywords 1309 may be, for example, a set of keywords and their combinations generated from the plurality of database objects 1304A, which includes one or more objects 1314A-1314ZZ. Keywords 1309 may also be "learned" from a user through

performing queries, or may be provided through a separate keyword administrator interface associated with query processor 1301.

Query processor 1301 also includes an interpreter and dictionary processor 1307 that receives objects identified by the reference dictionary object identifier 1305 and determines an optimal interpretation of the received objects. More specifically, processor 1307 determines optimal interpretations of the received objects, resolves ambiguities, updates information models 1311, and interacts with users to facilitate learning. Processor 1307 utilizes rules 1312 and heuristics 1313 to resolve ambiguities in determining the optimal interpretation of query 1302. Rules 1312 and heuristics 1313 may relate to information models 1311, which are in turn related to keywords 1309, cases 1310, and database objects 1304B in a semantic manner. When there are ambiguities in the interpretation of objects, e.g. multiple possible interpretations, multiple permissible combinations of meaningful objects, etc., rules 1312 and heuristics 1313 related to these objects are used to reduce or resolve these ambiguities.

Mapping processor 1306 performs a mapping between incoming objects and database objects 1304A. In particular, processor 1306 may generate database queries from the objects and the interpretations provided by identifier 1305 and processor 1307, respectively. Processor 1306, may, for example, generate SQL queries used to locate database objects 1304A. These queries may be executed by an SQL search engine, and processor 1301 may provide query result 1303 to user through, for example, a graphical user interface.

The literature on NLI clearly indicates that establishing a complete set of keywords is a key factor in handling ambiguity. However, according to one aspect of the invention, additional information beyond keywords are used to determine the meaning of the input query. This additional information makes it possible to use a collection of keywords far smaller than those required by the prior art. Particularly, there are four layers of resources comprising a data dictionary that are used to relate an incoming query 1302 to database object 1304A; i.e., cases 1310, keywords 1309, information models 1311, and database object values 1308B. These resources may be integrated through an extensible metadata representation method so that every piece of resources references to all other related resources in a semantically-based graphic. For instance, a keyword 1309 points to the semantic subject(s) it refers to, which points in turn to entities, relationships, and items pertaining to the subject(s), and ultimately to

database object values 1304B. The keywords 1309 also connect to cases 1310 involving them. The core of the reference dictionary (information model, initial keywords, and database structure) maybe, for example, a design-time product, developed by the analysts, designers, and users. Cases and additional keywords, metadata (e.g., changes to the information model) and database values may be added during operation of the system, and thus the system ages and evolves. A learning mechanism allows richer keywords and cases to provide more accurate performance. The reference dictionary enables a computer system to recognize a feasible region of interpretations of the input query 1302 and evaluate them. The reference dictionary 1308 also serves as the basis for interaction with the user (identifying needs and generating meaningful reference points) and acquisition of lessons (determining additional keywords and cases) - i.e., the reference dictionary may be used to assist the user in learning.

Reference dictionary has four fundamental attributes, as compared to conventional systems: it generates search-ready graphics-based representation of all four layers of resources; it supports learning; it simplifies keywords, and it assures complete interpretations of natural queries. Regarding the last two points, the inclusion of information models 1311 and case information 1310 reduces the volume of keywords 1309 needed to reduce the first two sources of ambiguity. For example, consider a natural articulation in the form of a short essay. If the essay consists of n words of which m are database objects or other recognized dictionary entries, there could be n/m words associated with each known term. These n/m words become the candidate keywords for the term. When including phrases (grouping of words), there could be, in theory, up to m*(n/m)! new keywords implied from the short essay. It is desired to increase the number m (hits) because the bigger m becomes, the fewer (exponentially) the possible groupings of words becomes, thus resulting in fewer new keywords to consider or to add to the dictionary.

Properly-developed information models 1311 having rich semantics provide a large m for the initial design of keywords, and increase the chance of subsequent "hits" (their use in queries) in practice resulting in less ambiguity, less possible interpretations to search, and less new keywords needed. Case information 1310 do not directly change m, but do help in resolving some ambiguity and hence still helps reducing the need for new keywords. Information models 1311 and cases 1310 represent a tightly structured, efficient kernel of meaning with which the users are familiar and tend to use more

frequently in their articulation with respect to the particular databases. In addition, information models 1311 and case information 1310 also contribute to resolving another type of ambiguity. In particular, they identify the possible missing information for incomplete input, by examining the graphics of the reference dictionary. Therefore, a reference dictionary determines more accurately and quickly than conventional systems a complete set of possible interpretations for queries articulated in a natural language format.

The basic logic of solution of the search and learn approach according to one embodiment of the invention will now be described. Given a reference dictionary **R** = **{C, K, M, D}**, representing respectively the sets of cases **C**, keywords **K**, information models **M**, and database values **D**, a method for searching according to one embodiment of the invention is as follows:

Step 1: Identify all words and phrases in the input natural language query 1302 that also belong to **R**. Denote this set of elements **I** (including possibly elements from **K, M or D**).

Step 2: Determine all possible, complete paths implied by **I** that span all input elements and query 1302 and belong to the overall graphics of **R**. These paths might include additional elements inferred from the reference dictionary in order to complete the paths. A complete path includes elements (original or inferred) in **M** and **D**. Each path corresponds to a particular interpretation of the original query.

Step 3: Search for the best interpretation by using some rules and heuristics of search. If multiple possible solutions exist, then use the elements in **C** that are associated with elements of **I** to resolve the ambiguity.

Step 4: Map the result to the database query language. Obtain the results of query and confirm them with the user.

Note that a learning mechanism may be engaged to interact with the user whenever the result provided at each step is insufficient. The outcome of the learning is stored in the system 1301 as new cases and keywords added to **C** and **K**, respectively. Also, note that each step allows for a wide range of possible strategies and algorithms to implement it.

Reference dictionary 1308 may also be based on the metadatabase model described in the aforementioned metadatabase references. In particular, a reference dictionary having a model that integrates four different types of enterprise metadata may be used. These metadata types include: database structure, semantic model, application, and software resource. The model may be used to form a core of the reference dictionary, and this core may be extended to include other three layers: keywords, cases and database values, and hence form the integrative (connected) structure of the reference dictionary. The other benefits of using this model includes its capability to incorporate rules and to support global query processing across multiple databases. A modeling system helps the development and creation of the metadatabase.

A structure of an example reference dictionary 1401 is shown in Figure 14. Each object in the figure represents either a table of metadata (in the case of square icon and diamond icon), or a particular type of integrity control rules (in the case of double diamond and broken diamond). These metadata include subjects and views, entity-relationship models, contextual knowledge in the form of rules, application and user definitions, database definitions and values, keywords, and cases.

Keywords, as noted above, are the natural words and phrases users use to refer to database objects and information model elements in natural articulation. They could represent instances, operators, items (attributes), entities, relationships, subjects, and applications. A keyword according to one embodiment of the invention is defined as an ordered pair of ***(class, object)***. Classes include Application, Subject, EntRel (entity-relationship), Item, Value, and Operator; all of which are metadata tables shown in Figure 14. Objects are instances (contents) of these classes. Because a hierarchy of objects in the core structure of the reference dictionary is Item-EntRel-Subject-Application, an object can be identified by an ordered quadruple (Item name, EntRel name, Subject name, Application name). In the model, however, each object has a unique identifier, thus the ordered quadruple is not needed to uniquely identify each object. It should be understood that any method for identifying objects may be used.

A case in case-based reasoning paradigm typically includes three components: problem definition, solution, and its outcome. New problems would use the problem definition to find the (best) matching cases and apply the associated solutions to them. The third component is useful when the domain knowledge is incomplete or unpredictable. In this research, the reference dictionary contains complete domain

knowledge needed, thus, we expand the problem definition but drop outcome. The system uses cases to resolve ambiguity in the recognition of meaningful terms (i.e., user's natural terms that are included in the reference dictionary) in the input and to help determine the solution among multiple possible interpretations. Thus, the case structure includes case-id, case-type, choices, context, and solution. For the type of resolving term ambiguities, a set of known terms describes the context (for problem definition). User's selection among possible choices of the meaningful term defines the solution. For the interpretation ambiguities type, a set of known elements of the information model describes the context, possible paths in the information model define the choices, and user's selection solution.

The resources (entries) of the reference model are connected in two ways. Recall that the structure shown in Figure 14 is a meta-schema representing the types and organization of all enterprise metadata. Thus, the elements of information models are metadata instances stored in some of the meta-entities (squares) and meta-relationships (diamonds) of the structure. These model elements are themselves connected internally in terms of their entity-relationship semantics. They are also connected externally to other types of resources including database values, keywords, and cases through the meta-schema. Keywords and cases are connected to information models and database values through particular meta-relationships. In other words, elements of information models (subjects, entities, relationships, and items) and keywords are linked to the database objects they represent. Therefore, the reference dictionary contains sufficient knowledge to determine the database objects involved and required for all queries defined sufficiently in information model elements or keywords.

Each sufficient statement corresponds to a complete and unique path (connection) of these elements and their corresponding database objects. (An SQL-like style database called MSQL may determine the shortest path when alternative paths exist. MSQL is discussed further in the journal entitled The Model-Assisted Global Query System for Multiple Databases in Distributed Enterprises, ACM Trans. Information Systems, 14:4, October 1996, pp. 421-470.) These complete paths represent the system's interpretations of users' queries. Ambiguity exists when a statement is insufficient such that there are conflicting interpretations - multiple paths leading to different database objects - for the query. These multiple paths could be the result either from providing incomplete

elements or from providing conflicting elements implied in the input, or both. Such are the cases easily taking place with truly natural articulation of database queries.

There are several different ways to handle ambiguity. First, the system employs a rich information model to maximize the chance with which the users would naturally choose its elements in their articulation. Second, the system uses keywords to capture the words in the natural articulation that the information model misses. It is worthwhile to reiterate that the information model is the roadmap (together with database values) for developing keywords at design time. These keywords represent multiple natural equivalents of terms used in the information model (and database values). As explained above, a rich information model not only lessens the burden of "scoring hit" on the keywords, it also greatly reduces the complexity of adding new keywords at the run time. Third, it accumulates cases of usage from actual operation and applies them to resolve remaining ambiguity when both information model and keywords are insufficient for a query. Interaction with the users is the last measure to sufficiently close the loop and finish the job. The NLI systematically involves users to provide the final resolution of ambiguity and confirmation of the result if needed. This learning also generates new cases and keywords and enhances the old cases.

Added to this basic logic is particular search strategies and additional search knowledge. Search includes the identification of all possible paths-interpretations (when ambiguity exists) and the evaluation of them. A search algorithm could follow a branch-and-bound strategy to minimize the space of search (limiting the number of possible paths to search). The development of bounds and branching rules would require a way to evaluate a given path with respect to the original natural query. A method for eliminating paths may also be used; that is, the system could infer contradiction based on the information model and perhaps operational rules (contextual knowledge) the reference dictionary contains. A method of optimization - inferring goodness of fit for the user - could be performed. Information about user's profile, concerned applications, and past cases are among the metadata that could be used form a basis to identify the most probable interpretations. Elimination is more conservative, but robust, than optimization because elimination places safety (correctness) first.

Three progressive levels of search and learn capabilities may be used. First, the system develops the most efficient way to enumerate all possible interpretations for a natural query (i.e., design a powerful reference dictionary). Second, it will develop

evaluation methods to improve the performance of search (i.e., eliminate more paths early in the search process) over the basic method. Finally, the system also proactively suggests the best interpretation for the user (i.e., develop case-based reasoning and other heuristics). Learning methods may accompany these search strategies at all levels. The above ideas are illustrated below with a brief example.

Consider a Computer-Integrated Manufacturing system consisting of three databases: order processing, process planning, and shop floor control. Suppose the system used the well-known TSER - Two Stage Entity-Relationship method (whose constructs include Application-Subject-Context-Entity-Relationship-Item described more in detail in the journal article entitled Paradigm Translations in Integrating Manufacturing Engineering Using a Meta-Model: the TSER Approach, by Cheng Hsu et al., J. Information Systems Engineering, 1:1, September 1993, pp. 325-352) to develop their information models and created a reference dictionary. These models became metadata instances stored in certain meta-entities and meta-relationships according to Figure 1. The system had also included proper keywords and cases and stored them along with all database objects. Now, suppose a user made the following sample request, a problem of NLI: An enterprise user of a Computer-Integrated Manufacturing database could query the system such as: "I have asked you many times and I still have not heard a clear answer. Listen, I would like to know the status of my order. I want to know what models you have started working on and how many of them you have completed. I placed this order, I believe, sometime around the 20th of last December. Could you please help? Thanks."

A text scanning and information retrieval algorithm (described further below) may generate the result shown in Table 1.

Terms	Possible Interpretations
status	(Item, SHOP_FLOOR.STATUS)
order	(Subject, ORDER) (EntRel, ORDER_PROCESSING.ORDER)
models	(EntRel, ORDER_PROCESSING.PART) (EntRel, SHOP_FLOOR.PART) (EntRel, PROCESS_PLAN.PART) (Subject,PART)
started working on	(Value, SHOP_FLOOR.STATUS)
completed	(Item, SHOP_FLOOR.NUM_COMPLETED)
20th of last December	(Value, ORDER_PROCESSING.DATE_DESIRED)

Table 1: Recognized Terms and Possible Interpretations

Ambiguity exists at terms "Order" and "Models" because each has multiple interpretations identified from the reference model. Otherwise, the result is definitive (forming a unique overall path) and could be mapped to the Metadatabase Query Language (MQL).

In this particular query, the information model would be sufficient to sort out the ambiguity and suggest a unique, optimal interpretation for these terms, and hence for the natural query. Still, cases could also be used either to confirm or to assist the resolution of ambiguity. However, there may be another kind of ambiguity in the input; the user indicated "around" 20th of last December in the original natural query. Because of this ambiguity, the user may find the final answer less than satisfactory. The system generally would have no method for interpreting correctly this piece of input since the user herself was ambivalent about it. There may be, in this instance, no proper solution other than to leaving the interpretation to the user. The final answer (based on 12/20/1999) may represent the best point estimation for the user's fuzzy interval of possibilities. The user would be presented with a chance to comment on this estimate and to request new dates for another query if wanted. Interaction with the user concerning this estimate, in its own right, could add valuable cases to the system so that it could provide more help for uncertain users when the term "around" is encountered in a following query, which is also a part of learning.

The following is an example algorithm for identifying meaningful terms:

```
Let m = number of words of input string
Let OrderedTerm = ()
Let n = 1
Do while (n <= m)
    Search in OrderedTerm for Input SubString starts at n-th position
    If (Term is found with length l),
    Then
        Let n = l + 1
    Else
        Search in the dictionary for Input SubString starts at n-th position
        If (Term is found with length l),
        Then
            Let Term = l-words string
            Retrieve meaning sets of the Term
            Insert OrderedTerm with the Term
            Let n = l + 1
        Else
            Let n = n + 1
        End if
    End if
End do while
```

The application of cases - i.e., matching a query with a case - is based on the vector space model as is known in the art. Two binary vectors represent a case (C) and a query (Q); and their COSINE measure indicates the goodness of fit. Below is an algorithm applying cases to the resolution of ambiguity in terms.

```
Retrieve cases containing the similar situation
If there are retrieved cases,
Then
    Let similar_value = 0
    For each retrieved case
            Let the base set for meaning space = set of meaning of terms of query
            For each meaning of case
                    Update the base set
            End for each
            Form a term space as an ordered n-tuples of terms /* n is a number of terms in the base set */
            Form a binary vector for query (Q) corresponding to the meaning space
            Form a binary vector for case (C) corresponding to the meaning space
            Compute COSINE similarity: COSINE (Q, C) = (Q.C)/(|Q|)(|C|)
            If COSINE (Q, C) > similar_value,
            Then
                    Let the solution case = the current case
                    Let similar_value = COSINE (Q,C)
            End if
    End for each
    If similar_value > accepted_value,
    Then
            Determine the meaning of the ambiguous term from the solution case
    End if
End if
```

Finally, when ambiguities are resolved, the complete list of path information becomes the sufficient input for the underlying database query language. In this

example, we show the MQL statements, which may function on top of a standard SQL facility.

```
FROM OE/PR WO_SEQ GET STATUS
FROM SUBJECT ORDER GET CUST_ORDER_ID, DATE_DESIRED,
   OD_STATUS, CUST_ID, ORDER_LINE_ID, PART_ID,
QUANTITY,
   DATE_SCHED, OI_STATUS, DESCRIPTION, COST
FROM OE/PR WK_ORDER GET NUM_COMPLETED
FOR STATUS = 'START WORKING ON'
AND DATE_DESIRED ='12/20/1999';
```

The mapping would perform processing in order to determine the GET lists and some conditions (such as AND/OR). However, at this point, the reference model would have all information needed to perform the query.

Having now described a few embodiments of the invention, it should be apparent to those skilled in the art that the foregoing is merely illustrative and not limiting, having been presented by way of example only. Numerous modifications and other embodiments are within the scope of ordinary skill in the art and are contemplated as falling within the scope of the invention as defined by the appended claims and equivalents thereto.

CLAIMS

1. A method for performing a transaction involving data provided by a first entity to a second entity over a distributed communications network, the method comprising:

maintaining the data, wherein the data is associated with the first entity; and

controlling distribution of the data to the second entity by the first entity.

2. The method according to claim 1, wherein the first entity is a person.

3. The method according to claim 1, wherein the first entity is a company.

4. The method according to claim 1, wherein the second entity is a computer system controlled by at least one of the group comprising:

a person;

an association of people;

a business; and

a group of businesses.

5. The method according to claim 1, wherein the first entity controls access information sufficient to perform a transaction.

6. The method according to claim 1, further comprising a step of controlling ownership, by the first entity, to the associated data.

7. The method according to claim 1, wherein the data associated with the first entity includes at least one preference of the first entity.

8. The method according to claim 7, wherein the at least one preference is determined by a transaction conducted by the first entity.

9. The method according to claim 1, further comprising a step of collecting, by the first entity, a history of one or more transactions performed by the first entity, and determining at least one preference of the first entity based on the history of one or more transactions.

10. The method according to claim 1, further comprising creating, as a result of a transaction, data owned by at least one of the group including:

the first entity;

the second entity; and

both the first and second entity.

11. The method according to claim 9, further comprising maintaining ownership information indicating ownership of the data.

12. The method according to claim 9, wherein the data is owned by both the first and second entity, and the ownership information indicates only joint ownership information.

13. The method according to claim 10, wherein the data includes at least one preference derived from data associated with the first entity.

14. The method according to claim 13, wherein the at least one preference is derived from data associated with other entities.

15. The method according to claim 14, wherein the other entities are other users in a user community, and wherein the data associated with other entities includes behavior and preference data associated with the other users.

16. A method for performing transactions over a distributed communications network involving at least one person, the method comprising:

maintaining, by the at least one person, data owned by the at least one person; and

controlling, by the at least one person, distribution of data at least one entity over the distributed network.

17. The method according to claim 16, further comprising a step of providing a subset of the data sufficient to perform a transaction.

18. The method according to claim 16, further comprising a step of controlling ownership of the data by the at least one person.

19. The method according to claim 16, further comprising a step of creating, as a result of executing a transaction, data owned by at least one of the group of:

the person;

the at least one entity; and

both the person and the at least one entity.

20. A method for maintaining ownership and control of data, comprising:

operating a computer system having a plurality of processes, and wherein at least one of the processes executes as a user process;

indicating, for data accessed by the at least one process, ownership of the accessed data; and

indicating, for the data accessed by the at least one process, control of the accessed data wherein indication of ownership and indication of control are independent.

21. The method according to claim 20, wherein an indication of ownership includes at least one of a group including:

a first person, wherein the data is personal data of the first person;

a first group, wherein the first person is a member of the first group and the first person accesses the data through the computer system;

automatic transfer of ownership of data, wherein a receiver of data including at least one of the first person or first group attains automatic ownership of the data; and wherein an indication of control includes at least one of a group including:

the first person;

a second person, wherein the second person accesses the data using a computer system;

the first group;

a second group wherein the second person is a member of the second group and accesses the data through the computer system; and

a system.

22. The method according to claim 21, wherein the indication of ownership further comprises indicating that the data has no owner.

23. The method according to claim 21, wherein at least one of the second person or the second group, upon receiving the data, attains ownership of the data.

24. The method according to claim 21, wherein at least one of the second person or second group attains ownership of the data based upon a predetermined relationship between an owner of the data and the at least one of the second person or the second group.

25. The method according to claim 24, wherein the predetermined relationship is an employer-employee relationship, wherein the owner is an employee, and the at least one of the second person or the second group is an employer.

26. The method according to claim 24, wherein the predetermined relationship is a legal relationship that obligates the owner of the data to relinquish ownership of the data to the at least one of the second person or the second group.

27. A method for maintaining ownership and control of data, comprising:

(a) operating, by a person, a computer system, wherein the computer system is configured to operate upon the data;

(b) indicating ownership of the data; and

(c) indicating, independently from (b) an indication of control of the data.

28. The method according to claim 27, wherein the steps (b) and (c) both include indicating, in the data, ownership and control of the data.

29. The method according to claim 27, further comprising transferring the data between first and second entities, and, as a result of a processing of the data, creating second data having an ownership by at least one of the group including:

the first entity;

the second entity; and

both the first and second entity.

30. The method according to claim 27, further comprising transferring the data between first and second entities, and providing control of the data by at least one of the group including:

the first entity;

the second entity; and

both the first and second entity.

31. The method according to claim 27, further comprising transferring the data between first and second entities, and wherein steps (b) and (c) both include indicating, in the data, ownership and control of the data.

32. A portable device for storing personal information of a user and conducting transactions using the personal information, the device comprising:

a database of personal information; and

an interface that transmits the personal information to an external entity to facilitate a transaction, wherein the interface controls what personal information is transmitted.

33. The device according to claim 32, wherein the personal information is personal preferences in goods and services, and the device interprets the personal information to determine marketing information to be presented to the user.

34. The device according to claim 32, wherein the interface further comprises a shell that permits or denies personal data to be distributed to an external entity.

35. The device according to claim 32, wherein the interface further comprises a shell that filters unwanted data from being received by the device.

36. The device according to claim 34, wherein a minimum amount of personal data is distributed to conduct a transaction with the external entity.

37. The device according to claim 34, wherein the shell is a software program executing on the portable device.

38. The device according to claim 35, wherein the shell is a software program executing on the portable device.

39. The device according to claim 32, further comprising a database of information relating to the use of the personal information.

40. The device according to claim 32, further comprising subsystems distributed in at least one of:

(a) one or more personal computers; and

(b) one or more network servers; and both (a) and (b).

41. The device according to claim 39, wherein the database of information relating to the use of the personal information includes at least one rule governing at least one of use access, and management of the personal information.

42. The device according to claim 41, wherein the database of information relating to the use of the personal information further comprises rules governing distribution of the personal information to external entities.

43. The device according to claim 42, further comprising a plurality of agents configured to support conducting transactions between a user of the device, wherein the plurality of agents control distribution of the personal information to the external entities.

44. The device according to claim 32, wherein the personal device cooperatively distributes the personal information among one or more trusted systems.

45. A method of accessing personal information of a user on a personal device, comprising steps of:

establishing a communication link between the device and an external system;

transferring the personal information to the external system to facilitate a transaction; and

storing information related to the transaction on the personal device.

46. The method according to claim 45, wherein the personal data includes medical records, and the information related to the transaction is medical treatment information.

47. The method according to claim 45, wherein the personal data includes financial records, and the information related to the transaction is financial transaction information.

48. The method according to claim 45, wherein the step of transferring includes transferring a minimum amount of personal data sufficient to conduct the transaction with the external system.

49. The method according to claim 45, further comprising a step of selecting personal data to be transferred by the step of transferring based on a minimum set of information needed to conduct the transaction.

50. The method according to claim 45, further comprising a step of storing, on a second external system, overflow personal information.

51. The method according to claim 45, further comprising steps of transferring the personal information to an intermediate system and generating an agent that performs the transaction on behalf of the user.

52. A method for managing personal data comprising:

storing, on a portable device, personal data owned by a user; and

controlling, by the user, distribution of the data to other entities over a communications network, wherein the personal data is distributed among the portable device and a plurality of general-purpose computers, the general-purpose computers being used to store overflow information.

53. The method according to claim 52, wherein at least one of the plurality of general-purpose computers is a personal computer operated by the user.

54. The method according to claim 52, further comprising filtering unwanted data from being received by the device.

55. The method according to claim 52, further comprising allowing personal data to be communicated to another entity, the communicated personal data being sufficient to support a transaction.

56. The method according to claim 54, wherein the communicated personal data is a minimum amount of data sufficient to support the transaction.

57. The method according to claim 54, further comprising denying, communication of information unnecessary for performing the transaction.

58. The method according to claim 52, wherein the step of controlling is performed cooperatively between the portable device and general-purpose computer.

59. The method according to claim 57, further comprising communicating the personal data with a server system, and generating, by the server system, an agent configured to control and transmit the personal data to the other entities.

60. The method according to claim 52, further comprising storing, on at least one of the plurality of general-purpose computers, personal data related to a community of which the user is associated.

61. The method according to claim 52, further comprising providing an interoperating distributed software system, the system being distributed among the portable device and plurality of general-purpose computers.

62. The method according to claim 52, further comprising providing, among the portable device and plurality of general-purpose computers, a common interface to the user.

63. The method according to claim 52, further comprising providing, among the portable device and plurality of general-purpose computers, a common interface to application software programs.

64. The method according to claim 52, further comprising storing, managing, and processing rules and agents to facilitate transaction conducted by the portable device and plurality of general-purpose computers.

65. A method for managing personal data comprising:

storing, on a first system, personal data owned by a user;

receiving and executing, at the first system, an externally-generated program configured to transfer information to second system; and

controlling, by the user, distribution of the personal data to the second system.

66. The method according to claim 65, wherein the externally-generated program is a software cookie.

67. The method according to claim 65, wherein the personal data owned by a user includes behavior information of the user.

68. The method according to claim 65, wherein the step of controlling further comprises allowing access, to the externally-generated program, to a minimum amount of personal data.

69. The method according to claim 65, wherein the step of controlling includes allowing a user to benefit from the externally-generated program while limiting an amount of personal data accessed by the program.

70. A method for performing a transaction in a distributed network comprising:

generating, by a computer system coupled to a communications network, a software agent configured to exchange data to support a transaction; and

exchanging the data with one or more other systems coupled to the communications network, wherein the agent provides a minimum amount of data to support the transaction.

71. The method according to claim 70, wherein the software agent is comprised of a plurality of base agents and the step of generating comprises generating the plurality of base agents that comprise the software agent.

72. The method according to claim 71, wherein at least one of the plurality of base agents is generated by another base agent.

73. The method according to claim 71, further comprising maintaining identification of each of the plurality of base agents and managing the identifications in a database.

74. The method according to claim 73, further comprising managing creation and deletion of agents.

75. The method according to claim 74, further comprising re-using existing agents to form new agents.

76. The method according to claim 70, wherein the computer system includes at least three devices functioning as a single manager of data including:

a portable device;

a system configured to connect to the portable device and store overflow information; and

a server system configured to dispatch the software agent to the one or more other systems coupled to the communication network.

77. A method for interfacing a portable device to a user comprising:

providing a natural language query input to the user;

performing, based on the input, a search of one or more language-based databases; and

providing, through an interface of the portable device, a result of the search.

78. The method according to claim 77, further comprising a step of identifying, for the one or more language-based databases, a finite number of database objects, and determining a plurality of combinations of the finite number of database objects.

79. The method according to claim 78, further comprising a step of mapping the natural language query to the plurality of combinations.

80. The method according to claim 79, wherein the step of mapping comprises steps of:

identifying keywords in the natural language query; and

relating the keywords to the plurality of combinations.

81. The method according to claim 79, further comprising a step of determining a reference dictionary comprising:

case information;

keywords;

information models; and

database values.

82. The method according to claim 79, wherein the step of mapping further comprises resolving ambiguity between the keywords and the plurality of combinations.

83. The method according to claim 82, wherein the step of resolving includes determining an optimal interpretation of the natural language query using at least one of a group comprising rules and heuristics.

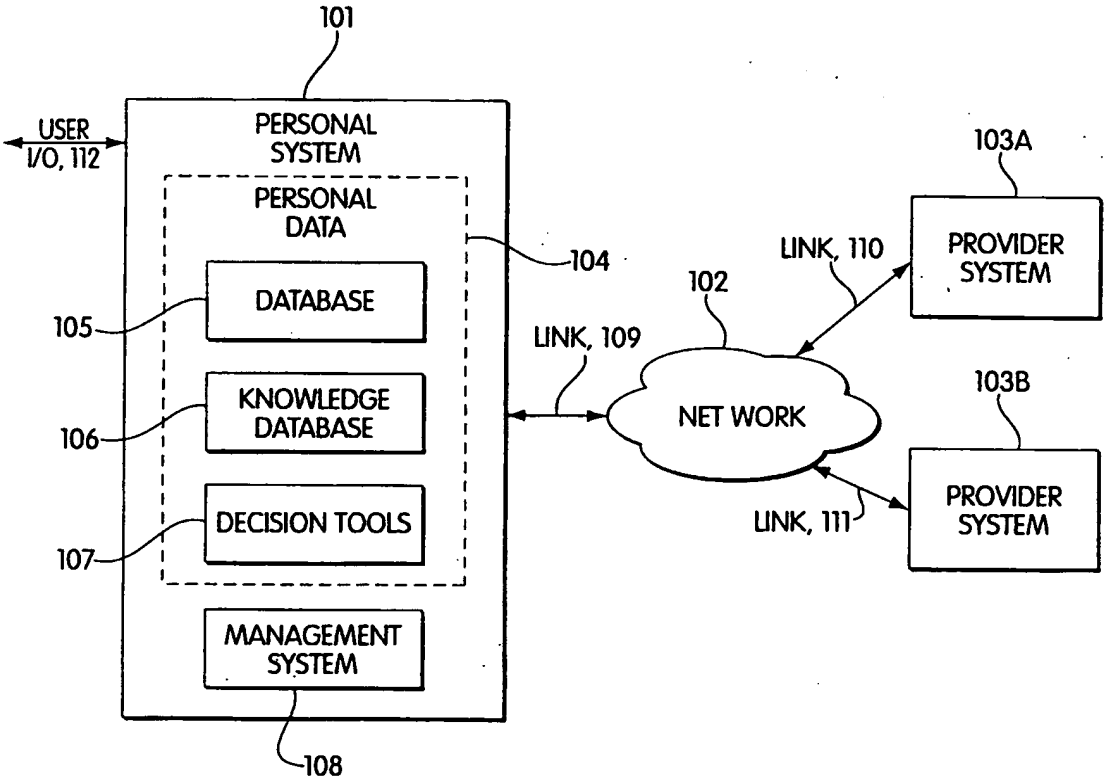

101
USER I/O, 112
PERSONAL SYSTEM
PERSONAL DATA
DATABASE
KNOWLEDGE DATABASE
DECISION TOOLS
MANAGEMENT SYSTEM
104
105
106
107
108
LINK, 109
102
NET WORK
LINK, 110
LINK, 111
103A
PROVIDER SYSTEM
103B
PROVIDER SYSTEM

Fig. 1

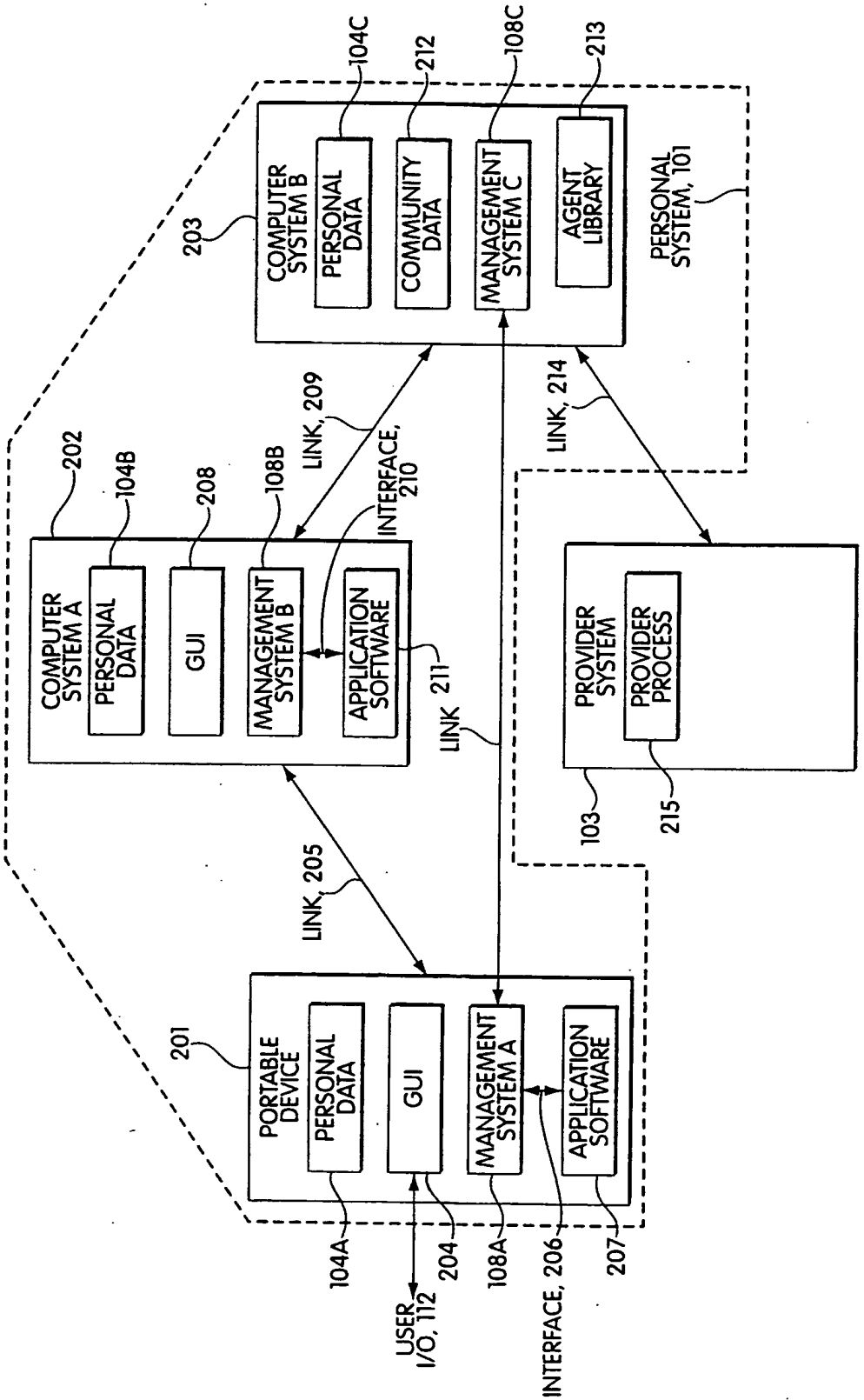

PERSONAL SYSTEM, 101
COMPUTER SYSTEM B
PERSONAL DATA
COMMUNITY DATA
MANAGEMENT SYSTEM C
AGENT LIBRARY
104C
212
108C
213
203
LINK, 209
INTERFACE, 210
LINK, 214
COMPUTER SYSTEM A
PERSONAL DATA
GUI
MANAGEMENT SYSTEM B
APPLICATION SOFTWARE
202
104B
208
108B
LINK, 211
PROVIDER SYSTEM
PROVIDER PROCESS
103
215
LINK, 205
PORTABLE DEVICE
PERSONAL DATA
GUI
MANAGEMENT SYSTEM A
APPLICATION SOFTWARE
201
104A
204
108A
207
USER I/O, 112
INTERFACE, 206

Fig. 2

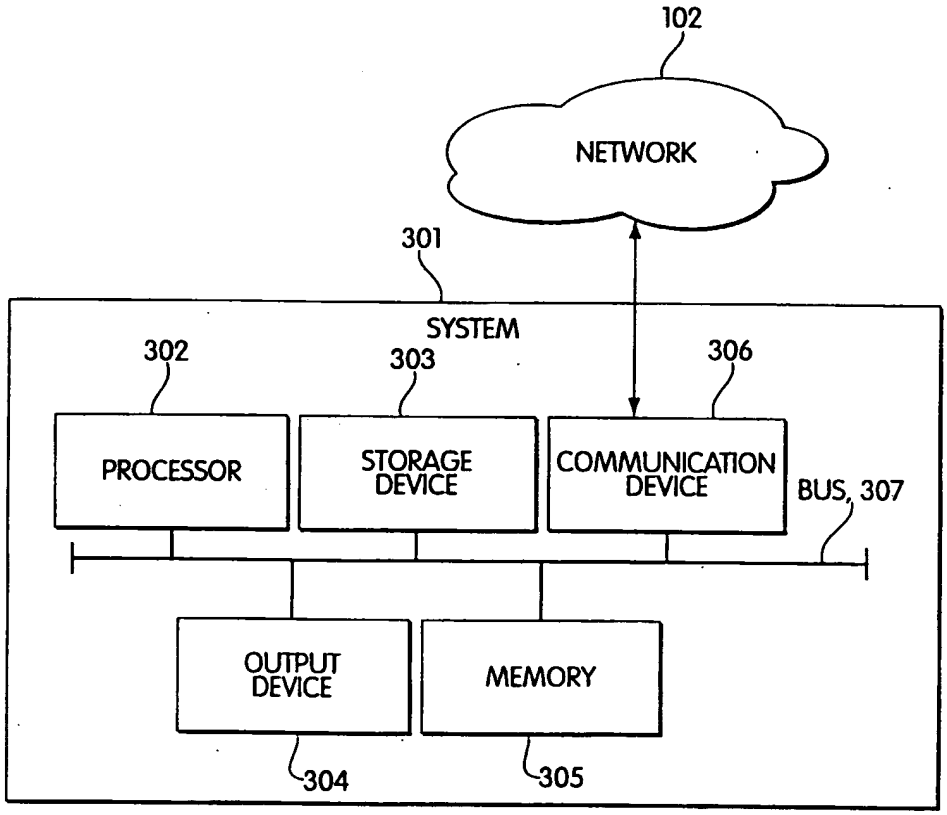

102
NETWORK
301
SYSTEM
302
303
306
PROCESSOR
STORAGE DEVICE
COMMUNICATION DEVICE
BUS, 307
OUTPUT DEVICE
MEMORY
304
305

Fig. 3

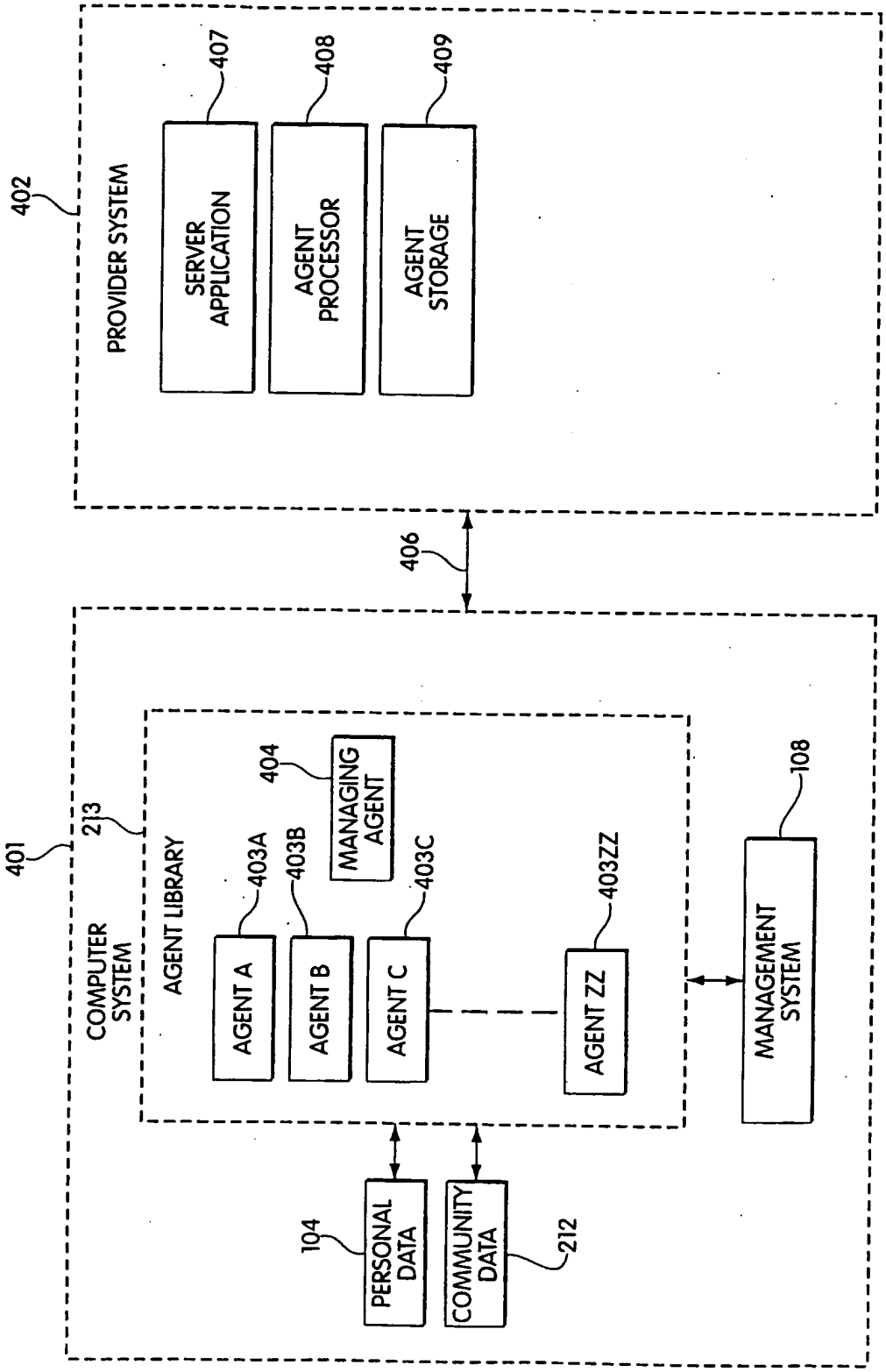

402
PROVIDER SYSTEM
407
SERVER APPLICATION
408
AGENT PROCESSOR
409
AGENT STORAGE
406
401
COMPUTER SYSTEM
213
AGENT LIBRARY
404
MANAGING AGENT
403A
AGENT A
403B
AGENT B
403C
AGENT C
403ZZ
AGENT ZZ
108
MANAGEMENT SYSTEM
104
PERSONAL DATA
212
COMMUNITY DATA

Fig. 4

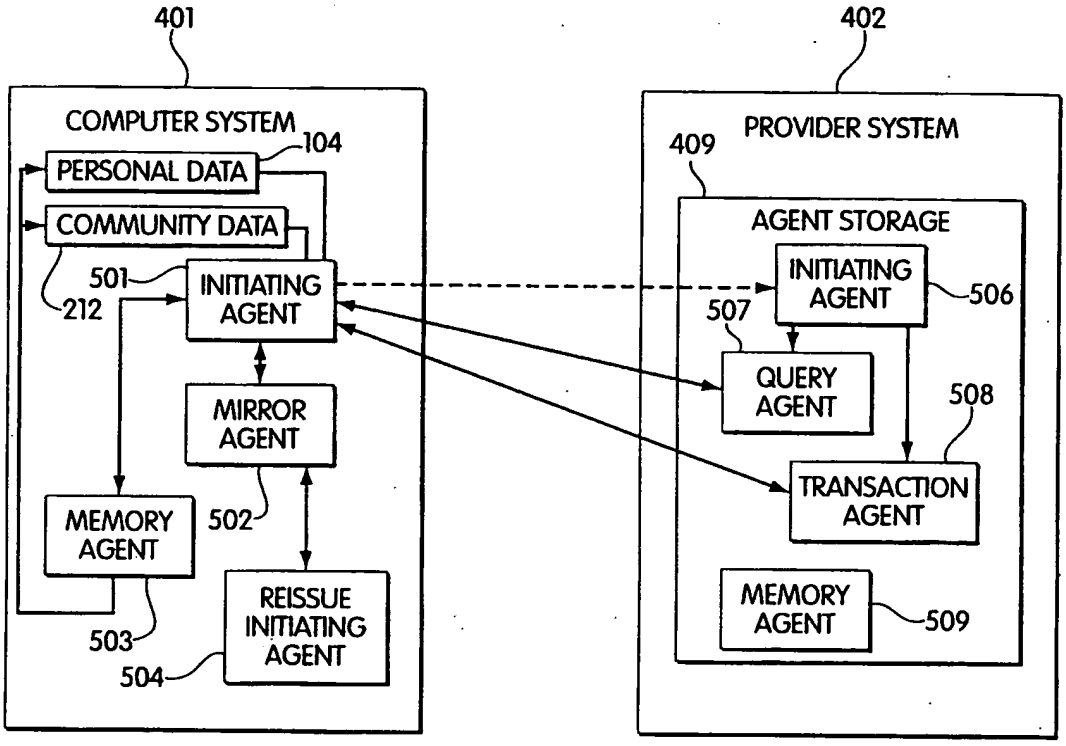

401
402
COMPUTER SYSTEM
104
PERSONAL DATA
COMMUNITY DATA
501
212
INITIATING AGENT
MIRROR AGENT
502
MEMORY AGENT
503
REISSUE INITIATING AGENT
504
PROVIDER SYSTEM
409
AGENT STORAGE
INITIATING AGENT
506
507
QUERY AGENT
508
TRANSACTION AGENT
MEMORY AGENT
509

Fig. 5

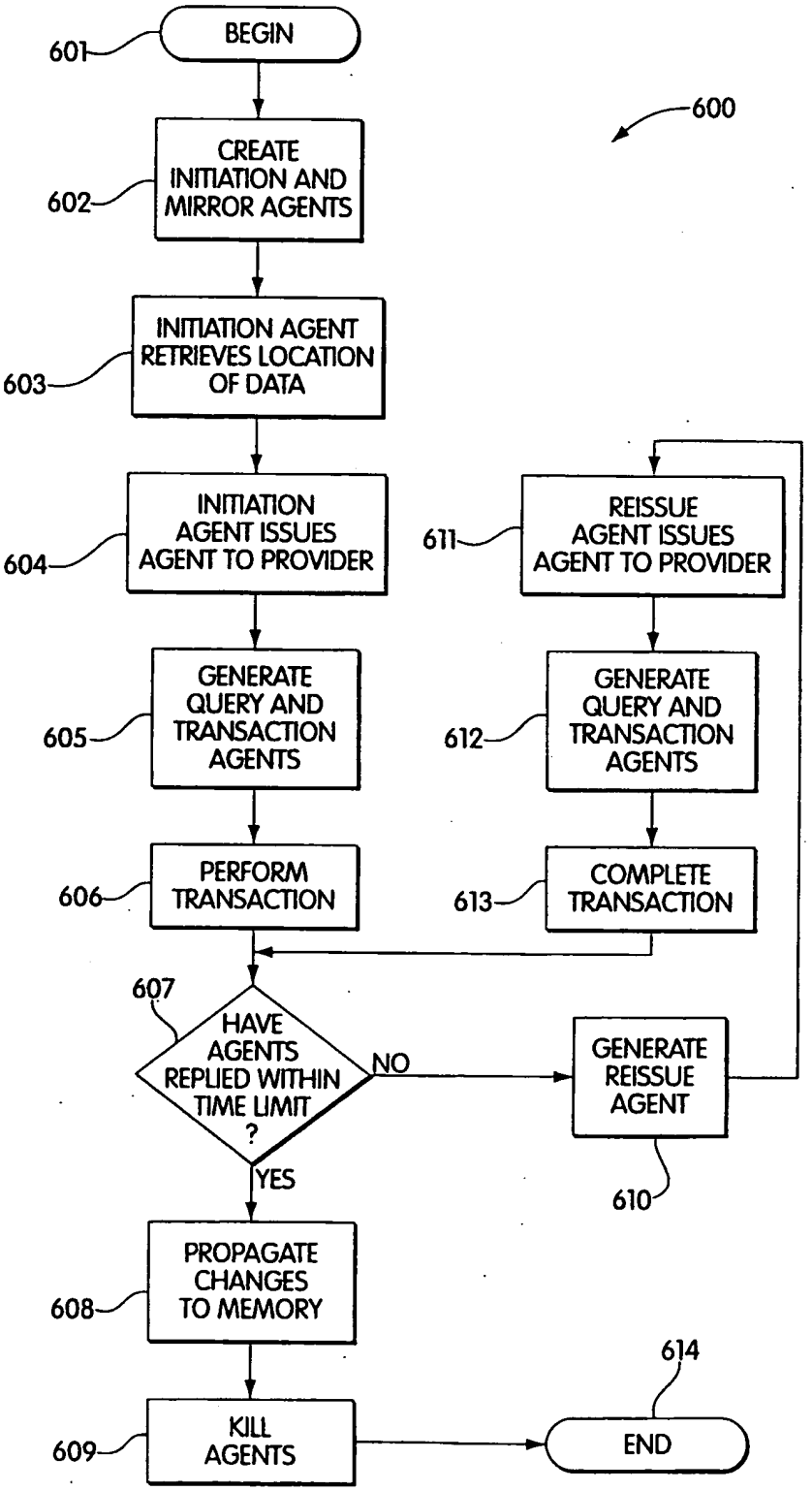

BEGIN
601
600
CREATE INITIATION AND MIRROR AGENTS
602
INITIATION AGENT RETRIEVES LOCATION OF DATA
603
INITIATION AGENT ISSUES AGENT TO PROVIDER
604
REISSUE AGENT ISSUES AGENT TO PROVIDER
611
GENERATE QUERY AND TRANSACTION AGENTS
605
GENERATE QUERY AND TRANSACTION AGENTS
612
PERFORM TRANSACTION
606
COMPLETE TRANSACTION
613
607
HAVE AGENTS REPLIED WITHIN TIME LIMIT ?
NO
YES
GENERATE REISSUE AGENT
610
PROPAGATE CHANGES TO MEMORY
608
614
KILL AGENTS
609
END

Fig. 6

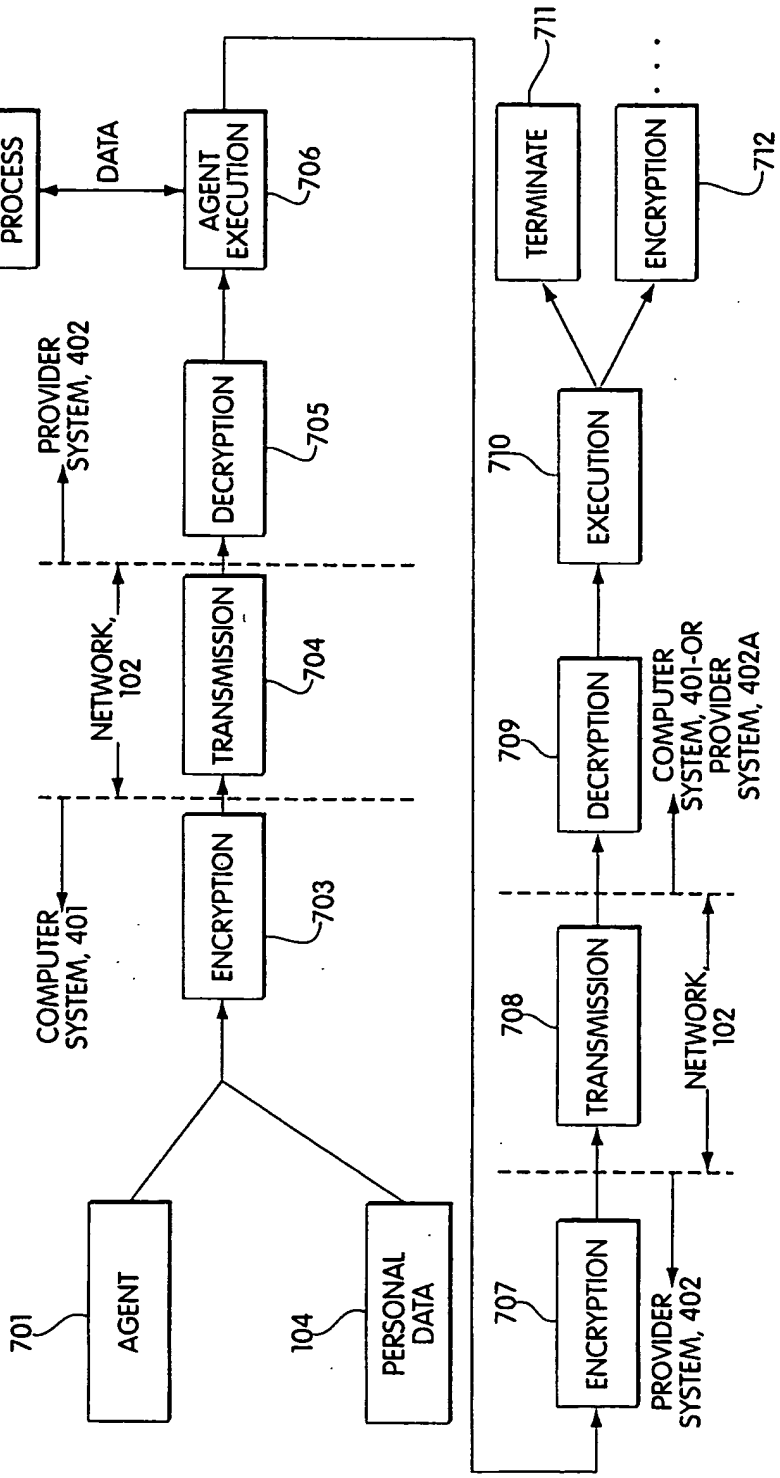

AGENT
701
PERSONAL DATA
104
ENCRYPTION
703
TRANSMISSION
704
DECRYPTION
705
AGENT EXECUTION
706
PROVIDER PROCESS
215
DATA
COMPUTER SYSTEM, 401
NETWORK, 102
PROVIDER SYSTEM, 402
ENCRYPTION
707
TRANSMISSION
708
DECRYPTION
709
EXECUTION
710
TERMINATE
711
ENCRYPTION
712
PROVIDER SYSTEM, 402
NETWORK, 102
COMPUTER SYSTEM, 401-OR PROVIDER SYSTEM, 402A

Fig. 7

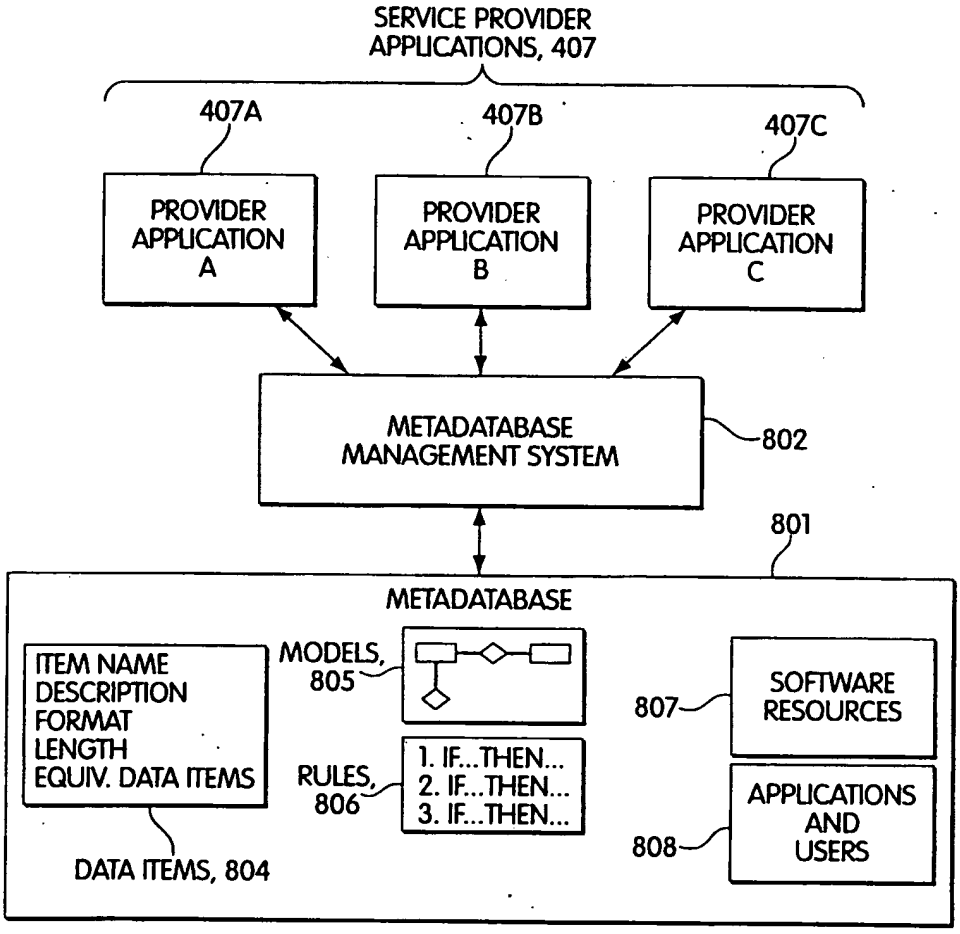

SERVICE PROVIDER APPLICATIONS, 407
407A
407B
407C
PROVIDER APPLICATION A
PROVIDER APPLICATION B
PROVIDER APPLICATION C
METADATABASE MANAGEMENT SYSTEM
802
801
METADATABASE
ITEM NAME
DESCRIPTION
FORMAT
LENGTH
EQUIV. DATA ITEMS
DATA ITEMS, 804
MODELS, 805
RULES, 806
1. IF...THEN...
2. IF...THEN...
3. IF...THEN...
807
SOFTWARE RESOURCES
808
APPLICATIONS AND USERS

Fig. 8

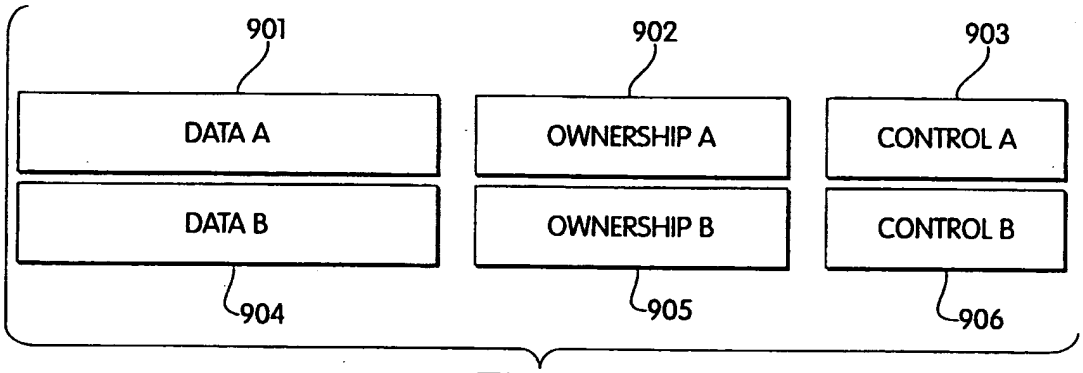

901
902
903
DATA A
OWNERSHIP A
CONTROL A
DATA B
OWNERSHIP B
CONTROL B
904
905
906

Fig. 9

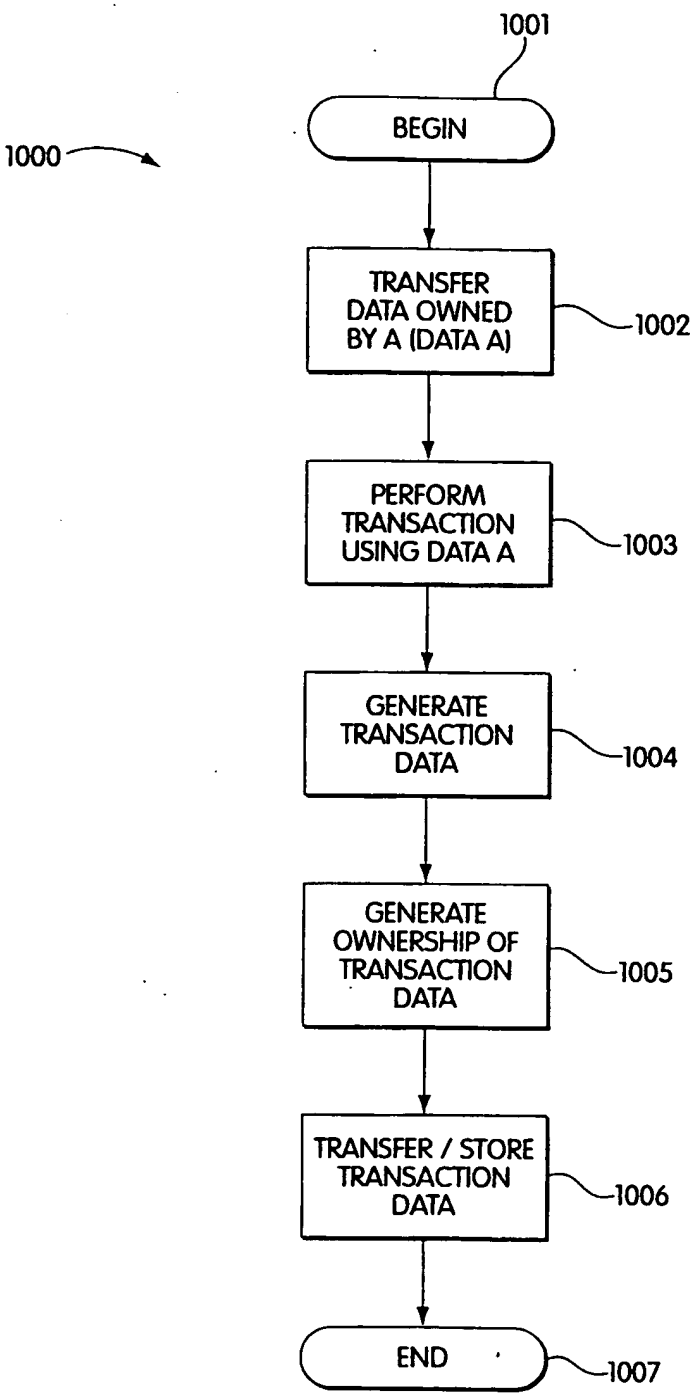

1000
1001
BEGIN
TRANSFER DATA OWNED BY A (DATA A)
1002
PERFORM TRANSACTION USING DATA A
1003
GENERATE TRANSACTION DATA
1004
GENERATE OWNERSHIP OF TRANSACTION DATA
1005
TRANSFER / STORE TRANSACTION DATA
1006
END
1007

Fig. 10

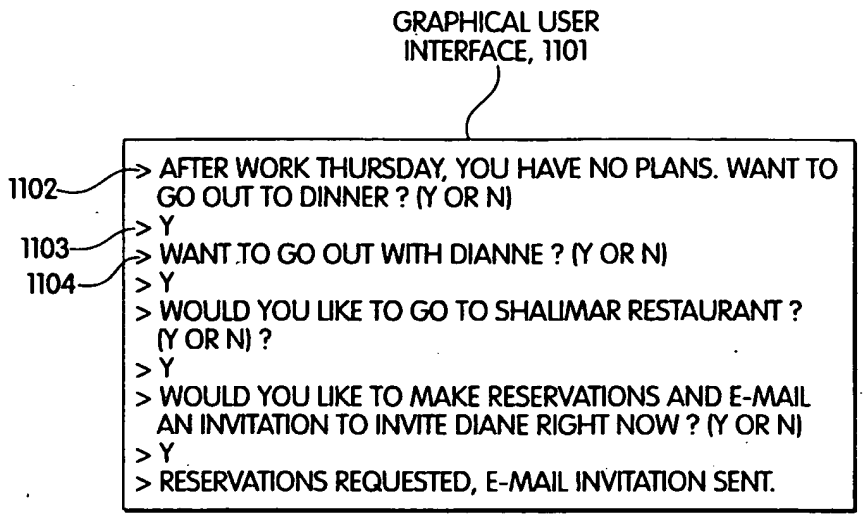

GRAPHICAL USER INTERFACE, 1101
1102
1103
1104
> AFTER WORK THURSDAY, YOU HAVE NO PLANS. WANT TO GO OUT TO DINNER ? (Y OR N)
> Y
> WANT TO GO OUT WITH DIANNE ? (Y OR N)
> Y
> WOULD YOU LIKE TO GO TO SHALIMAR RESTAURANT ? (Y OR N) ?
> Y
> WOULD YOU LIKE TO MAKE RESERVATIONS AND E-MAIL AN INVITATION TO INVITE DIANE RIGHT NOW ? (Y OR N)
> Y
> RESERVATIONS REQUESTED, E-MAIL INVITATION SENT.

Fig. 11

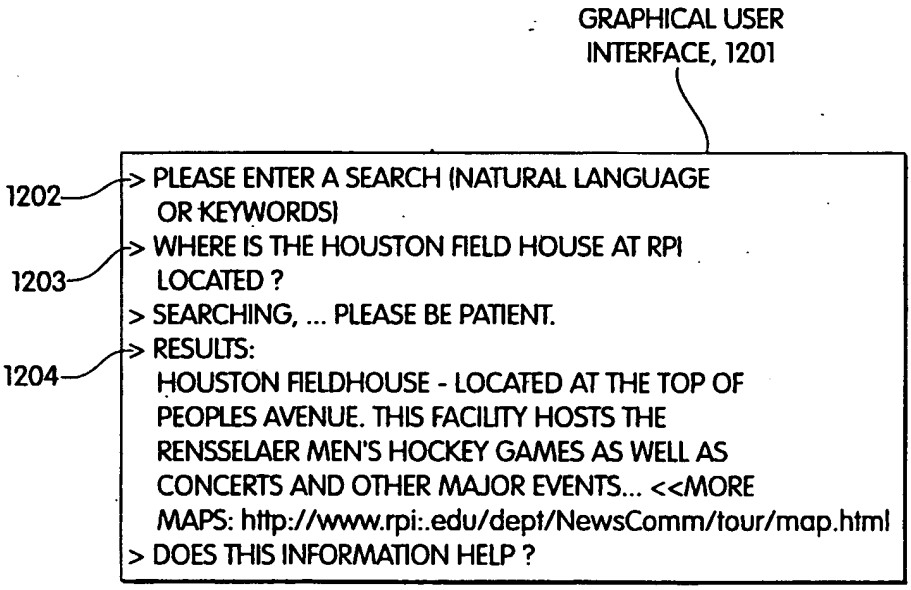

GRAPHICAL USER INTERFACE, 1201
1202
1203
1204
> PLEASE ENTER A SEARCH (NATURAL LANGUAGE OR KEYWORDS)
> WHERE IS THE HOUSTON FIELD HOUSE AT RPI LOCATED ?
> SEARCHING, ... PLEASE BE PATIENT.
> RESULTS:
HOUSTON FIELDHOUSE - LOCATED AT THE TOP OF PEOPLES AVENUE. THIS FACILITY HOSTS THE RENSSELAER MEN'S HOCKEY GAMES AS WELL AS CONCERTS AND OTHER MAJOR EVENTS... <<MORE
MAPS: http://www.rpi:.edu/dept/NewsComm/tour/map.html
> DOES THIS INFORMATION HELP ?

Fig. 12

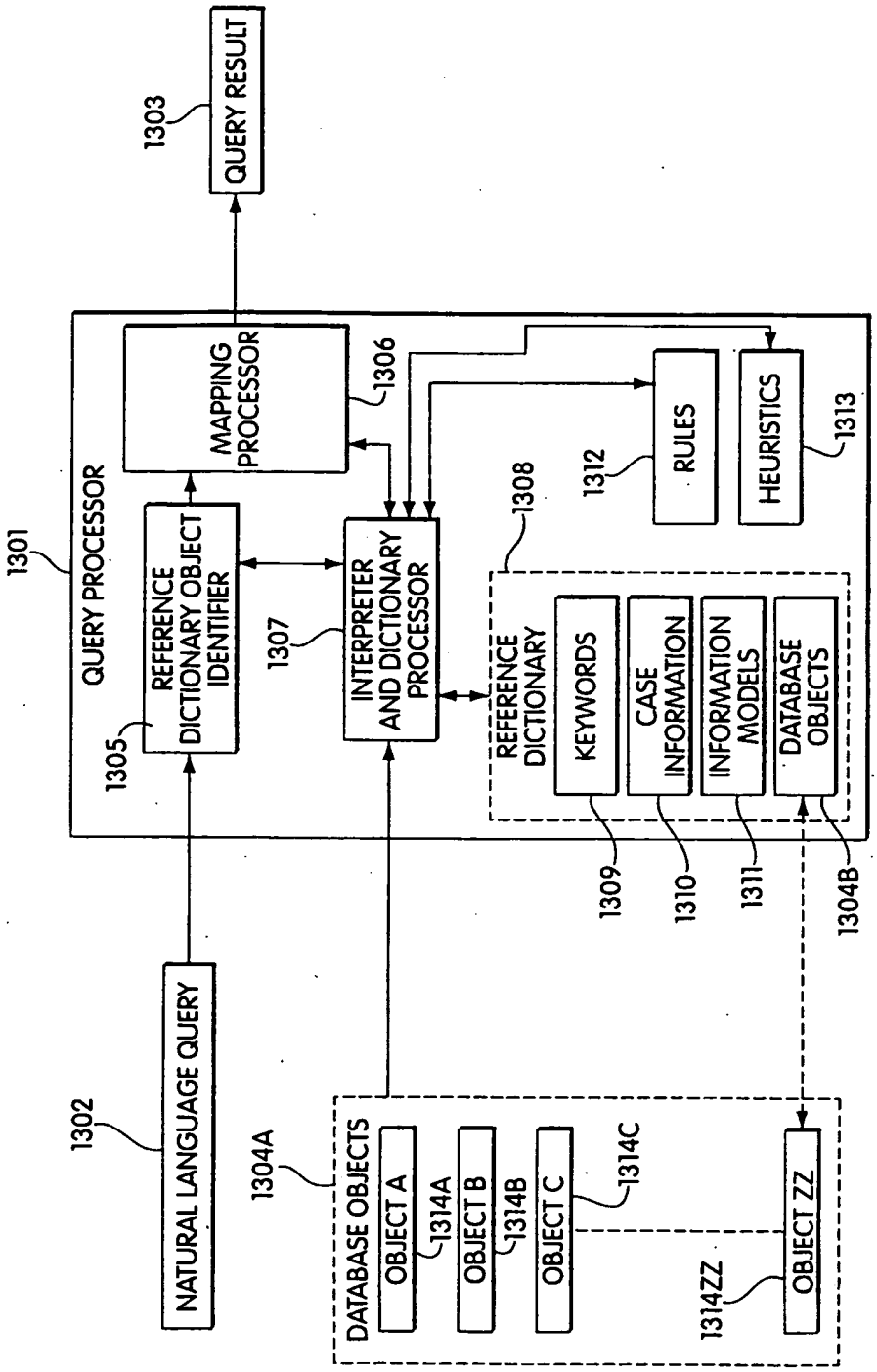

1303
QUERY RESULT
1301
QUERY PROCESSOR
MAPPING PROCESSOR
1306
REFERENCE DICTIONARY OBJECT IDENTIFIER
1305
1307
INTERPRETER AND DICTIONARY PROCESSOR
1308
1312
RULES
HEURISTICS
1313
REFERENCE DICTIONARY
KEYWORDS
CASE INFORMATION
INFORMATION MODELS
DATABASE OBJECTS
1309
1310
1311
1304B
1302
NATURAL LANGUAGE QUERY
1304A
DATABASE OBJECTS
OBJECT A
1314A
OBJECT B
1314B
OBJECT C
1314C
1314ZZ
OBJECT ZZ

Fig. 13

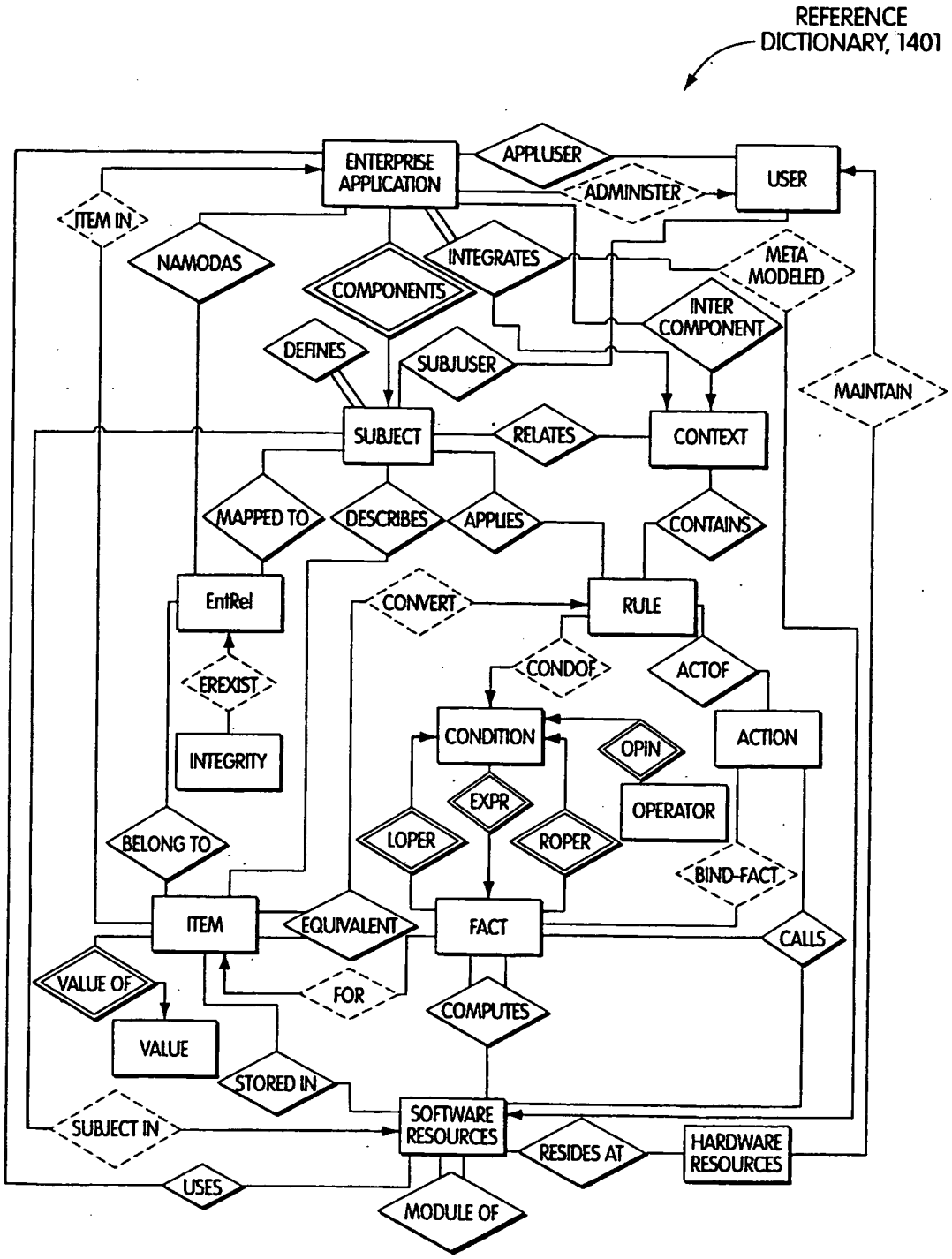

REFERENCE DICTIONARY, 1401
ENTERPRISE APPLICATION
APPLUSER
USER
ADMINISTER
ITEM IN
NAMODAS
INTEGRATES
COMPONENTS
META MODELED
INTER COMPONENT
DEFINES
SUBJUSER
MAINTAIN
SUBJECT
RELATES
CONTEXT
MAPPED TO
DESCRIBES
APPLIES
CONTAINS
EntRel
CONVERT
RULE
CONDOF
ACTOF
EREXIST
CONDITION
OPIN
ACTION
INTEGRITY
EXPR
OPERATOR
BELONG TO
LOPER
ROPER
BIND-FACT
ITEM
EQUIVALENT
FACT
CALLS
VALUE OF
FOR
COMPUTES
VALUE
STORED IN
SUBJECT IN
SOFTWARE RESOURCES
RESIDES AT
HARDWARE RESOURCES
USES
MODULE OF

Fig. 14